



Universidad Carlos III de Madrid

Grado en Ingeniería Informática

Curso 2017-18

Trabajo de Fin de Grado

Desarrollo de un Planificador Visual Interactivo

**Para niños con Discapacidad Cognitiva
en Clases de Educación Especial**

Diego Alejandro González Ordóñez

Tutor

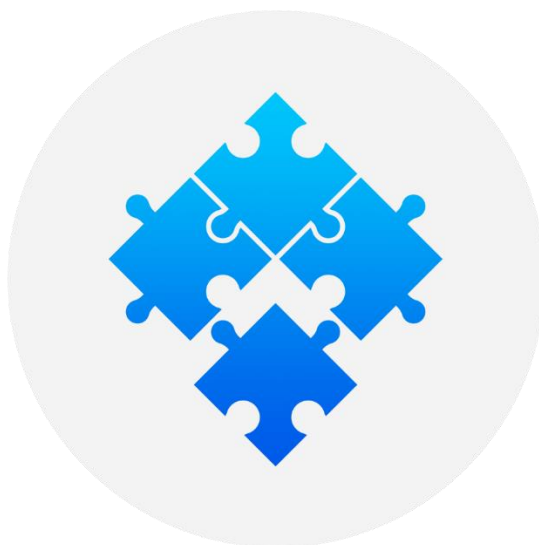
Andrea Bellucci

Leganés, 19 de junio de 2018



Esta obra se encuentra sujeta a la licencia Creative Commons

RESUMEN



La presente memoria detalla todo el trabajo realizado de cara al desarrollo de una aplicación cuyo objetivo es facilitar la vida a los educadores y padres de niños en clases de educación especial. La aplicación tiene un propósito muy concreto y es el de permitir la creación de planificadores visuales con pictogramas, también llamados “anticipadores de actividades”.

El problema que se plantea es la necesidad que tienen los niños con discapacidades cognitivas (por ejemplo, trastornos del espectro autista) de productos de apoyo para conocer con anticipación actividades fuera de rutina. El no disponer de un recurso con el que preparar actividades fuera de rutina les genera una gran cantidad de estrés que deriva en comportamientos atípicos como berrinches, autolesiones, agresiones físicas y verbales...

Por esta razón, los educadores preparan anticipadores de actividades por medio de herramientas de carácter general lo que supone una gran carga de tiempo y esfuerzo. El objetivo es proporcionar a educadores, padres y madres una herramienta fácil de usar con la que elaborar planificadores visuales por medio de una aplicación móvil multiplataforma.

Palabras clave: Discapacidades Cognitivas, Trastornos del Espectro Autista, Planificadores Visuales, Pictogramas, Aplicación Móvil.

AGRADECIMIENTOS

Primero, gracias a Alberto por su colaboración, que ha sido tan importante en este proyecto. Su trabajo con los niños y su predisposición para mejorar las cosas han sido una inspiración para mí.

A mi tutor Andrea Bellucci, gracias por confiar en mí y dejarme trabajar en este TFG.

Gracias a mis compañeros de la universidad, UC3M Pals, ellos han conseguido sacar lo mejor de mí durante todos estos años.

Por último, a mis padres, que siempre me han brindado su apoyo incondicional. Ellos son mi mayor ejemplo de trabajo, constancia, y compromiso. Gracias por todo lo que hacéis por mí.

ÍNDICE DE CONTENIDOS

1. INTRODUCCIÓN	1
1.1. Contexto	1
1.2. Problemas	2
1.3. Objetivos	3
1.4. Estructura del documento	4
2. ESTADO DEL ARTE	5
2.1. Trastorno del Espectro Autista	5
2.2. Productos de Apoyo y Sistemas Aumentativos y Alternativos de Comunicación	5
2.3. Planificadores Visuales	6
2.3.1. Finalidad	6
2.3.2. Beneficios de la utilización de planificadores visuales	7
2.3.3. Funcionamiento	7
2.4. Punto de vista tecnológico	8
2.5. Alternativas para la implementación	16
2.5.1. Aplicación en código nativo (Objective-C, Java)	16
2.5.2. Aplicación Híbrida (Cordova, Ionic)	16
2.5.3. Aplicación compilada a código nativo (React native)	17
3. ANÁLISIS DEL SISTEMA	18
3.1. Identificación de Usuarios	18
3.2. Problemas de los educadores	19
3.3. Análisis de la metodología actual	19
3.3.1. Microsoft Word	20
3.3.2. PowerPoint	22
3.4. Definición del Sistema	25
3.5. Requisitos	26
3.5.1. Requisitos Funcionales	27
3.5.2. Requisitos Modificados	34
3.5.3. Requisitos No Funcionales	40
4. DISEÑO Y JUSTIFICACIÓN DE LA SOLUCIÓN	42
4.1. Prototipo inicial	42
4.2. Implementación en Ionic Framework	47
4.2.1. Modelo	47
4.2.2. Vista	48
4.2.3. Controlador	48
4.3. Prototipo en ionic	49
4.4. Reunión con el educador	52

4.4.1. Entrevista.....	52
4.4.2. Prueba de la aplicación.....	54
4.5. Segundo prototipo	55
4.6. Pruebas y trabajo futuro	58
5. MARCO REGULADOR.....	59
5.1. Legislación Aplicable.....	59
5.2. Responsabilidades Éticas	60
5.3. Propiedad Intelectual.....	60
6. PLANIFICACIÓN DEL PROYECTO.....	61
6.1. Definición de Actividades	61
6.1.1. Planteamiento	61
6.1.2. Desarrollo.....	61
6.1.3. Documentación.....	61
6.2. Lista de actividades	62
6.3. Diagrama de GANT	63
7. ENTORNO SOCIOECONÓMICO.....	64
7.1. Costes de personal.....	64
7.1.1. Planteamiento	64
7.1.2. Desarrollo.....	65
7.1.3. Documentación.....	65
7.1.4. Costes de personal totales	65
7.2. Costes materiales.....	66
7.2.1. Costes Hardware	66
7.2.2. Costes Software.....	66
7.3. Resumen de costes y totales	67
8. CONCLUSIONES.....	68
9. RESUMEN EN INGLÉS. ABSTRACT.....	69
9.1. Introduction.....	69
9.2. State of the Art	70
9.3. System Analysis	75
9.4. System Implementation.....	77
9.5. Conclusions	78
10. BIBLIOGRAFÍA	79
11. GLOSARIO	81
ANEXO I. Consentimiento Informado para la entrevista.	83

ÍNDICE DE ILUSTRACIONES

Ilustración 1: Ejemplo de anticipadores con pictogramas	2
Ilustración 2: Ejemplo de planificador simple.....	3
Ilustración 3: Logotipo Araword	8
Ilustración 4: Araword, captura 1	9
Ilustración 5: PictogramAgenda logo	10
Ilustración 6: PictogramAgenda, captura 1	11
Ilustración 7: CPA logo	12
Ilustración 8: CPA, captura 1	13
Ilustración 9: LetMeTalk logo	13
Ilustración 10: LetMeTalk, captura 1	14
Ilustración 11: Proceso en Word, captura 1	20
Ilustración 12: Proceso en Word, captura 2.....	20
Ilustración 13: Proceso en Word, captura 3.....	21
Ilustración 14: Proceso en Word, captura 4.....	21
Ilustración 15: Proceso en Word, captura 5.....	21
Ilustración 16: Proceso en Word, captura 6.....	22
Ilustración 17: Proceso en PowerPoint, captura 1	23
Ilustración 18: Proceso en PowerPoint, captura 2	23
Ilustración 19: Proceso en PowerPoint, captura 3	23
Ilustración 20: Proceso en PowerPoint, captura 4	24
Ilustración 21: Prototipo inicial, Pantalla principal	43
Ilustración 22: Prototipo inicial, Pantalla de creación (I).....	43
Ilustración 23: Prototipo inicial, Pantalla de creación (II).....	44
Ilustración 24: Prototipo inicial, Pantalla de búsqueda	45
Ilustración 25: Prototipo inicial, Modo de visualización (I).....	46
Ilustración 26: Prototipo inicial, Modo de visualización (II)	46
Ilustración 27: Prototipo en ionic. Pantalla Principal	49
Ilustración 28: Prototipo en ionic. Creación de un planificador.....	50
Ilustración 29: Prototipo en ionic, Selección de pictogramas	51
Ilustración 30: Prototipo en ionic, Modo de visualización del planificador.....	51
Ilustración 31: Modo de visualización de pictogramas	52
Ilustración 32: Prototipo en ionic (II). Ventana de creación	55
Ilustración 33: Prototipo en ionic (II). Creación de plantilla.....	56
Ilustración 34: Prototipo en ionic (II). Creación desde plantilla	56
Ilustración 35: Prototipo en ionic (II). Colores de un grupo	57
Ilustración 36: Prototipo en ionic (II). Visualización de pictogramas modificado	58
Ilustración 37: Diagrama de GANTT	63
Ilustración 38: Ionic final prototype screenshots.....	77

ÍNDICE DE TABLAS

Tabla 1: Comparativa de aplicaciones actuales	15
Tabla 2: Plantilla de Requisitos Software.....	27
Tabla 3: Requisito Funcional.....	27
Tabla 4: Requisito Funcional 02.....	28
Tabla 5: Requisito Funcional 03.....	28
Tabla 6: Requisito Funcional 04.....	29
Tabla 7: Requisito Funcional 05.....	29
Tabla 8: Requisito Funcional 06.....	30
Tabla 9: Requisito Funcional 07.....	30
Tabla 10: Requisito Funcional 08.....	31
Tabla 11: Requisito Funcional 09.....	31
Tabla 12: Requisito Funcional 10.....	32
Tabla 13: Requisito Funcional 11.....	32
Tabla 14: Requisito Funcional 12.....	33
Tabla 15: Requisito Funcional 13.....	33
Tabla 16: Requisito Funcional 14.....	33
Tabla 17: Requisito Funcional 15.....	34
Tabla 18: Requisito Funcional 16.....	35
Tabla 19: Requisito Funcional 17.....	35
Tabla 20: Requisito Funcional 18.....	36
Tabla 21: Requisito Funcional 19.....	36
Tabla 22: Requisito Funcional 20.....	37
Tabla 23: Requisito Funcional 21.....	37
Tabla 24: Requisito Funcional 22.....	37
Tabla 25: Requisito Funcional 01 Modificado	38
Tabla 26: Requisito Funcional 09 Modificado	39
Tabla 27: Requisito Funcional 07 Modificado	39
Tabla 28: Requisito No Funcional 01	40
Tabla 29: Requisito No Funcional 02.....	40
Tabla 30: Requisito No Funcional 03.....	40
Tabla 31: Requisito No Funcional 04.....	41
Tabla 32: Horas de trabajo. Planteamiento.....	64
Tabla 33: Horas de trabajo. Desarrollo.....	65
Tabla 34: Horas de trabajo. Documentación	65
Tabla 35: Horas de trabajo. Totales.....	65
Tabla 36: Costes de personal. Totales	65
Tabla 37: Costes hardware	66
Tabla 38: Costes software.....	66
Tabla 39: Costes totales.....	67
Tabla 40: Comparative between apps.....	73
Tabla 41: Project budget.....	77

1. INTRODUCCIÓN

Los niños con Trastornos del Espectro Autista (TEA) habitualmente tienen dificultades con los cambios de rutina y transiciones entre actividades. Esto hace que desarrollen una fuerte dependencia hacia sus cuidadores (educadores y padres).

Para reducir esta dependencia existen soportes visuales como planificadores de actividades con pictogramas (Banda & Grimmer, 2008). Los profesores en clase de educación especial y los padres de niños con TEA hacen uso de estos soportes visuales, pero necesitan invertir una gran cantidad de tiempo para elaborarlos.

En el caso de las clases de educación especial, los planificadores visuales son un recurso útil para definir las actividades diarias en clase que constituyen la rutina de un alumno. Los educadores utilizan estos soportes visuales para reforzar la comprensión del niño con respecto a las tareas pasadas, presentes y futuras promoviendo así un comportamiento independiente. Sin embargo, hay ocasiones en las que es necesario planificar actividades fuera de la rutina diaria, como, por ejemplo, excursiones escolares. Si estas actividades fuera de rutina no se anticipan de manera adecuada, pueden generar trastornos en el comportamiento de los niños.

Para este tipo de actividades fuera de rutina, los educadores elaboran anticipadores, estos son planificadores visuales con pictogramas que incluyen información respecto a qué, cómo y cuándo se realizará la actividad futura. Con un anticipador se consigue reducir el impacto y el estrés que se produce con este cambio en la rutina, puesto que se trata de un acontecimiento que se anuncia con antelación durante las clases. Estos anticipadores además incluyen todas las acciones y tareas que se realizarán durante la actividad y sirven para reducir la dependencia de los niños en otras personas.

1.1. Contexto

La utilización de estos anticipadores es diaria, tanto en clase como en casa. Los niños con TEA están acostumbrados a una rutina y necesitan conocer con semanas de antelación las actividades que se van a realizar fuera de esta rutina. El no disponer de estos anticipadores les genera una gran cantidad de estrés que deriva en una mala aceptación de las actividades a realizar, llegando a provocar comportamientos agresivos física y verbalmente, autolesiones, poca disposición a colaborar, berrinches, etc. (Banda and Grimmer, 2008).

Hoy en día tenemos a disposición una gran variedad de dispositivos tecnológicos que nos facilitan la vida simplificando la utilización y reduciendo el tiempo que invertimos en ciertas tareas, como por ejemplo en comunicaciones, comercio electrónico, creación de contenido, etc. Sin embargo, actualmente no existe ninguna herramienta que venga a solucionar los problemas logísticos y de tiempo que tienen los educadores a la hora de elaborar los planificadores con pictogramas para los niños en clase de educación especial.



Ilustración 1: Ejemplo de anticipadores con pictogramas

Hay aplicaciones disponibles a las que se le podría dar un uso parecido, pero que no funcionan bien o no son fáciles de utilizar. En este TFG se pretende abordar este problema mediante la metodología de diseño centrado en el usuario.

1.2. Problemas

Actualmente, los planificadores se hacen con herramientas de propósito general como Microsoft Word y el proceso es lento y difícil. Estos problemas representan una pérdida de tiempo y desvían la atención de los educadores de la parte importante, que es pensar cómo preparar los anticipadores para que los niños lo entiendan de la mejor manera posible.

Por otra parte, los padres de estos niños pueden no disponer de las capacidades técnicas necesarias para utilizar estos programas y elaborar sus propios planificadores o modificar planificadores preparados por los educadores. Además, crear un buen planificador requiere de cierta experiencia. Es importante definir cuándo se llevará a cabo la actividad, con quién se realizará la actividad, el que se hará durante la actividad, que situaciones pueden darse, etc. Además, las secuencias de pictogramas deben de tener una precisión adecuada en cuanto a las acciones que se llevarán a cabo. Los padres pueden no disponer

del conocimiento necesario para elaborar un buen planificador, puesto que no existen plantillas o sugerencias a la hora de realizar un planificador.



Ilustración 2: Ejemplo de planificador simple

Por otra parte, cuando llega el día de la actividad a realizar, los educadores elaboran un “planificador simple”. Se trata de un acordeón con los pictogramas más significativos de la actividad. Los educadores los llevan consigo el día de la actividad.

Su elaboración es completamente manual y los educadores necesitan de papel, tijeras y mucho tiempo para elaborar estos acordeones. Estos planificadores simples sirven para guiar a los niños en todo momento para transitar entre actividades. Es importante que estos sean flexibles, puesto que, dependiendo de la situación, es importante mostrar el pictograma anterior, el actual o el de actividades posteriores.

1.3. Objetivos

El objetivo de este TFG es proporcionar una herramienta fácil de utilizar para la creación y la visualización de anticipadores para educadores y padres. De esta manera se puede facilitar la vida de los educadores y además se proporciona una herramienta que los padres pueden usar sin tener un conocimiento técnico ni una amplia formación para ayudar a sus hijos con el contenido que se puede crear con la aplicación.

A continuación, se resumen los principales objetivos que se pretenden alcanzar con este TFG. Mediante una aplicación, los objetivos que deberían quedar cubiertos son:

- Proporcionar una herramienta para generar planificadores con pictogramas que sea fácil de usar.

- Permitir la visualización de los planificadores, con la posibilidad de seleccionar pictogramas clave para su posterior visualización.
- Permitir que los planificadores creados puedan ser exportados y compartidos.

Estos objetivos son de carácter general, pero cada uno de estos objetivos se divide en diferentes metas más pequeñas. Es importante no perder de vista que lo que se intenta procurar por encima de todo es la facilidad de uso.

1.4. Estructura del documento

Para concluir la introducción, se detalla la estructura del presente documento. Este escrito se divide por capítulos.

En el siguiente capítulo, estado del arte, se estudia la literatura existente respecto a los distintos campos en los que se involucra este trabajo.

A continuación, en el capítulo 3, se realiza un análisis del sistema, identificando primero a los usuarios finales y especificando requisitos software que detallan las capacidades de la aplicación a desarrollar.

En el capítulo 4, se describen las decisiones de diseño y la implementación de la aplicación por medio de los distintos prototipos que se llevan a cabo.

Después, en el capítulo 5, se incluye un análisis del marco regulador que incluye aquellas normas y leyes aplicables al desarrollo del presente trabajo. Además, se tratan aspectos como responsabilidades éticas y propiedad intelectual.

Tras esto, en el capítulo 6, se detalla la planificación del proyecto, explicando las distintas fases que lo componen.

En el capítulo 7, se estudia el entorno socioeconómico del proyecto, incluyendo una estimación de los costes del mismo.

A continuación, en el capítulo 8, se vuelcan las conclusiones a las que se llega tras la finalización del presente trabajo, incluyendo un análisis de los resultados y planteando el trabajo futuro.

En el capítulo 9, se incluye un resumen del trabajo en inglés.

Por último, se incluyen dos capítulos que contendrán la bibliografía en la que se apoya este trabajo de fin de grado y un glosario de términos y abreviaturas.

2. ESTADO DEL ARTE

En este apartado se intenta dar una visión general de todo lo que se ha hecho hasta ahora en este campo a través de la literatura existente. El objetivo es identificar y realizar una síntesis del trabajo realizado hasta ahora en este tema, indagar dónde existe posibilidad de mejora y buscar alternativas a explorar.

2.1. Trastorno del Espectro Autista

Un buen comienzo sería definir qué son los Trastornos del Espectro Autista (TEA). Se trata de un trastorno neurobiológico cuyos síntomas principales son “deficiencias persistentes en la comunicación e interacción social, además de patrones restrictivos y repetitivos de comportamiento, intereses o actividades” (American Psychiatric Association, 2013). No se habla de autismo, sino que se habla de un espectro autista, puesto que tiene diferentes formas de presentarse y los casos que se dan son muy heterogéneos. Este trastorno afecta solo en España a alrededor de trescientas mil personas, de los cuales cincuenta mil están en edad escolar (Europapress, 2016).

La habilidad de comunicación es de vital importancia para poder desenvolverse en la sociedad y poder relacionarse con otras personas. Este trabajo se va a centrar en los problemas de comunicación que sufren estas personas.

A partir de estos problemas para comunicarse surge la necesidad del desarrollo de productos de apoyo y sistemas aumentativos y alternativos de comunicación.

2.2. Productos de Apoyo y Sistemas Aumentativos y Alternativos de Comunicación

Los productos de apoyo se definen como “cualquier producto (incluyendo dispositivos, equipamiento, instrumentos, tecnología y software), fabricado especialmente, o disponible en el mercado, para prevenir, compensar, controlar, mitigar o neutralizar deficiencias, limitaciones de la actividad o restricciones en la participación” (UNE-EN ISO 9999, 2011). Entre los productos de apoyo más usados para los niños con discapacidades cognitivas o TEA se encuentran los tableros de comunicación que utilizan en su día a día en clase, además, existen otro tipo de dispositivos como comunicadores electrónicos y software para ordenadores o tablets.

Por otra parte, los Sistemas Aumentativos y Alternativos de Comunicación (SAAC) son formas de expresarse diferentes al lenguaje hablado que tienen como objetivo mejorar o

compensar las deficiencias que sufren las personas con discapacidad. En estos SAAC, se utilizan sistemas de símbolos con productos de apoyo (ARASAAC, 2018).

Un ejemplo de SAAC sería un sistema pictográfico, que permite establecer comunicación directa con personas que no saben leer o tienen dificultades para ello. Estos sistemas pictográficos son flexibles y pueden adaptarse a un rango de personas con diferentes niveles de discapacidad cognitiva. Para establecer esta comunicación, los sistemas pictográficos se basan en el uso de pictogramas. Aunque existan diversos tipos de planificadores visuales como se puede leer en el artículo de Davies, C. (2008), este TFG se centrará en planificadores visuales con pictogramas.

2.3. Planificadores Visuales

Un planificador visual es un sistema pictográfico que se compone de una serie de pictogramas ordenados.

Estos pictogramas pueden presentarse de diversas maneras como, por ejemplo, en tiras de pictogramas de longitud variable o de manera individual.

Además, pueden ser de distintos tipos como texto, números, dibujos, fotos, etc. El tipo de contenido a utilizar en cada planificador dependerá de cada caso, por ejemplo, hay niños que entienden mejor los planificadores visuales con fotos de elementos reales y otros entienden mejor los dibujos.

2.3.1. Finalidad

Existen artículos como el de Takanori Koyama & Hui-Ting Wang (2011), en los que se hace un análisis de la utilización de planificadores visuales. En este artículo en concreto, se estudia el uso de planificadores visuales de actividades para promover la independencia de personas con autismo. Además, se habla de los problemas que se pretenden resolver y los beneficios que conlleva el uso de este tipo de planificadores.

En cuanto a los problemas que se mencionan en el artículo, se podrían destacar los siguientes:

- La dependencia de los niños con TEA respecto a personas adultas supervisoras.
- Aparición de conductas agresivas, poca disposición a colaborar, autolesiones, inconformidad en general cuando se transita entre actividades.

Estos mismos problemas también se mencionan en el artículo de Banda & Grimmett (2008).

2.3.2. Beneficios de la utilización de planificadores visuales

Un planificador visual de actividades sirve para afrontar los problemas mencionados, reduciendo así la dependencia de las personas con discapacidad cognitiva respecto a otras personas supervisoras. No se consigue eliminar la dependencia en supervisores por completo, ni se produce una total desaparición de las conductas atípicas y comportamientos disruptivos, pero se reducen en gran medida.

Lo que se consigue es promover una conducta más independiente y autodeterminada. En adición a esto, se pueden listar además otros beneficios que se pueden observar con la utilización de planificadores visuales:

- Los planificadores visuales se pueden adaptar a diferentes tipos de personas con diferentes discapacidades cognitivas.
- No son exclusivos para personas con TEA, sino que también se pueden aplicar para personas con otro tipo de discapacidades como, por ejemplo, síndrome de Down.
- Los planificadores visuales se pueden utilizar tanto en el colegio como en casa.
- Son aplicables a diferentes ámbitos y temáticas como juegos, actividades en la escuela, tareas de casa, horarios, etc.

2.3.3. Funcionamiento

En el artículo de Banda & Grimmer (2008) se habla del uso de estos soportes visuales como una estrategia educativa prometedora.

En el mismo artículo también se tratan los principios por los que estos soportes visuales funcionan. Estos son:

- La preferencia de los estímulos visuales por encima del resto puesto que este tipo de estímulo son la principal fuente de información para las personas con TEA.
- En adición a esto, prefieren la utilización de objetos por encima de recibir órdenes de una persona. En muchos casos, una persona con TEA tiende a distraerse con los rasgos físicos de la persona por encima de las cosas que esta persona dice.

Además, este tipo de soporte visual no requiere de un gran esfuerzo para su aprendizaje y utilización. Los participantes en el estudio del artículo aprenden a utilizar los planificadores visuales de manera independiente en clase.

2.4. Punto de vista tecnológico

Este tipo de soporte visual se puede integrar en dispositivos móviles como smartphones y tablets. Hay otro artículo en el que Davies, Stock y Wehmeyer (2002) hacen un análisis de los beneficios en la utilización de dispositivos electrónicos y llegan a la conclusión de que, con dispositivos electrónicos portátiles, por ejemplo, palmtops, se reduce el uso de indicaciones externas para la realización de tareas, incrementando así la independencia y autodeterminación en personas con discapacidades mentales.

El artículo mencionado es de 2002, pero ¿qué avances se han hecho en el uso de dispositivos electrónicos en este campo? A continuación, se hará mención de distintas aplicaciones presentes para dispositivos móviles y se realizará un análisis de las características más importantes para este TFG. El objetivo es extraer características comunes, tanto positivas como negativas, identificando así carencias, apartados donde mejorar y apartados que funcionan y que sería conveniente tener en cuenta.

En la Play Store se pueden encontrar muchas aplicaciones relacionadas con el autismo, pero este análisis se va a centrar en aplicaciones relacionadas con planificadores visuales, en concreto aquellas que sean populares y tengan valoraciones positivas. Las aplicaciones elegidas han sido Araword, PictogramAgenda, CPA, y LetMeTalk. Se han elegido porque tienen buenas valoraciones en la Play Store y, además, aparecen como aplicaciones recomendadas en la página web de ARASAAC.

Araword



Ilustración 3: Logotipo Araword

Se trata de un comunicador con pictogramas cuya principal funcionalidad es la de autocompletar con pictogramas las palabras que se escriben. Esto presenta ventajas, como la rapidez para crear un tablero con pictogramas. Sin embargo, por otra parte, existen varios pictogramas para representar una misma situación o acción y no hay la posibilidad de elegir cuál con precisión. Se trata de una aplicación responsiva, válida tanto para móviles como para tablets.

No presenta ningún tipo de tutorial, los botones no son del todo indicativos y el usuario debe suponer que hacen algunos botones de la interfaz. El diseño de la aplicación es simple, puesto que no tiene demasiadas funcionalidades. Hay funcionalidades que no son obvias para el usuario, como, por ejemplo, alternar el pictograma asociado a una palabra mediante una pulsación sobre el pictograma. Otra funcionalidad que no se ve a simple vista o que no se puede intuir es la de cambiar un pictograma por una imagen propia o cambiar el texto mediante una pulsación larga sobre el pictograma.

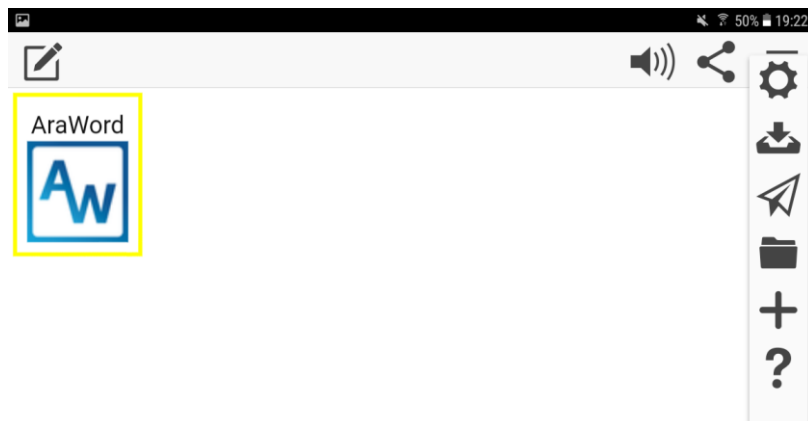


Ilustración 4: Araword, captura 1

En esta aplicación, se permite al usuario añadir imágenes propias y asignar a la imagen una palabra clave. Cuando el usuario escribe la palabra, la imagen aparece automáticamente.

La aplicación también permite compartir el tablero actual en formato de imagen. También se permite guardar el tablero actual en un formato que puede interpretar la propia aplicación.

La aplicación no tiene un buscador de pictogramas puesto que los pictogramas se añaden automáticamente a partir del texto. Esto puede ser una limitación para determinadas palabras, por ejemplo, si una palabra está mal escrita, o una palabra no tiene un pictograma asociado. Si una palabra no existe, lo normal es buscar un pictograma similar que pueda encajar con la palabra, en este caso, no es posible puesto que no hay un buscador como tal.

La aplicación no dispone de una forma de ver cada pictograma como secuencia, pero sí tiene un lector de texto que lee en voz alta las palabras asociadas a cada pictograma presente en el tablero en orden.

PictogramAgenda



Ilustración 5: PictogramAgenda logo

Se trata de una aplicación que permite crear una agenda visual con la que anticipar actividades. El principal problema de esta aplicación es una interfaz de usuario muy anticuada y muy poco intuitiva. Funciona tanto para móviles como para tablets, aunque la interfaz no es responsiva.

Utiliza los pictogramas proporcionados por ARASAAC y también permite la adición de imágenes propias. Esto se hace a través de un menú. Por otra parte, la aplicación está limitada a doce pictogramas.

El tablero de la agenda se puede compartir en un formato que solo puede interpretar la propia aplicación. De la misma forma se pueden guardar y cargar agendas.

La aplicación no incluye ninguna indicación de su funcionamiento y sería necesario. El diseño de la aplicación está anticuado, no es fácil de usar, requiere de exploración por parte del usuario. Los botones aparecen repetidos, y el icono elegido en muchos casos no indica exactamente su acción asociada. La aplicación permite hacer cosas más superfluas como cambiar el fondo del tablero, pero no permite realizar acciones más básicas como eliminar o cambiar un pictograma de manera rápida. Por último, cabe destacar también que la aplicación tiene bugs visuales al rotar la pantalla y parpadeos de la interfaz a la hora de cambiar de pictograma en modo secuencia.

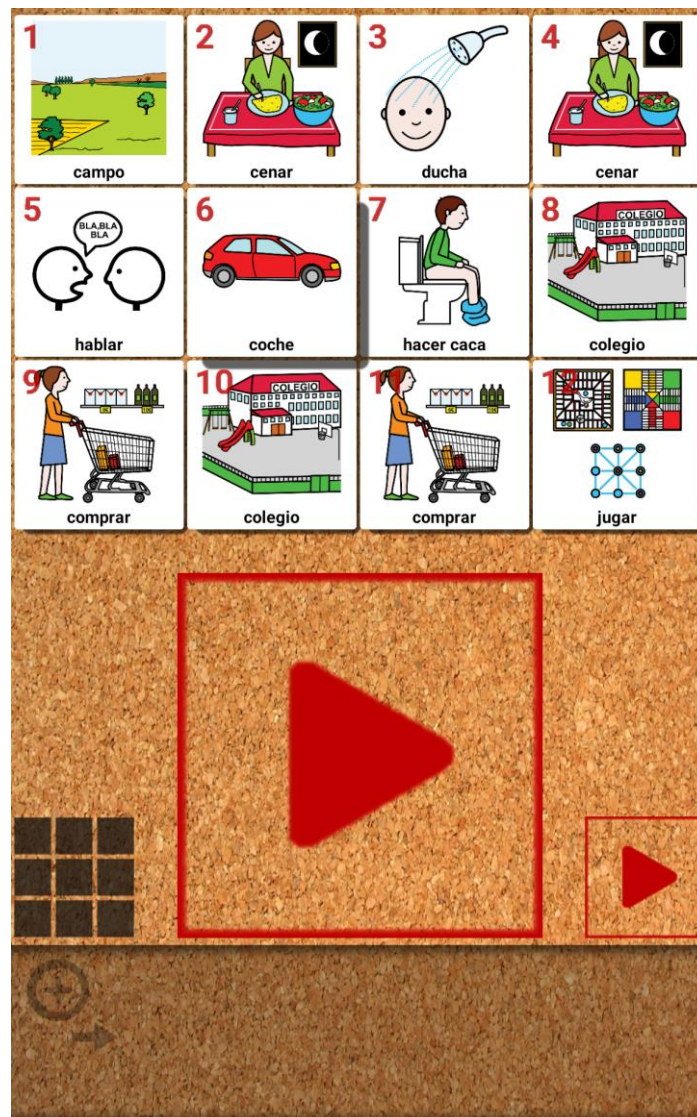


Ilustración 6: PictogramAgenda, captura 1

La aplicación no dispone de un buscador de pictogramas como tal, hace uso de menús y un explorador de archivos para la búsqueda de pictogramas. Esto es así tanto para los pictogramas de ARASAAC como para las imágenes personales.

Esta aplicación sí que dispone de una funcionalidad para mostrar los pictogramas a modo de secuencia. También permite ir adelante o atrás y ver que actividades ya se han completado.

CPA



Ilustración 7: CPA logo

Es un Comunicador Personal Adaptable. Sirve para personas con problemas graves de comunicación. Con esta aplicación se pueden crear secuencias de pictogramas con frases. La aplicación funciona para móviles y para tablets. También dispone de una versión para iOS.

La aplicación hace uso de los pictogramas de ARASAAC y también permite la adición de fotos o imágenes propias desde la galería.

Esta aplicación carece totalmente de una manera de compartir las tiras de pictogramas creadas.

La aplicación dispone de un tutorial para explicar las funcionalidades de la aplicación. Se trata de un tutorial muy largo ya que la aplicación difícil de usar y de entender.

El diseño de la aplicación deja que desear. En algunos apartados de la aplicación el contraste entre el color del fondo y la tipografía dificultan mucho la lectura, por ejemplo, un fondo azul oscuro con letras de color negro. La aplicación no es para nada intuitiva, el diseño en general es confuso puesto que existen dos zonas distintas donde añadir pictogramas. Además, hay decisiones de diseño cuestionables como botones que desaparecen tras un temporizador para cambiar de modo. En general, el diseño de la aplicación es malo o no se ha probado o estudiado lo suficiente.

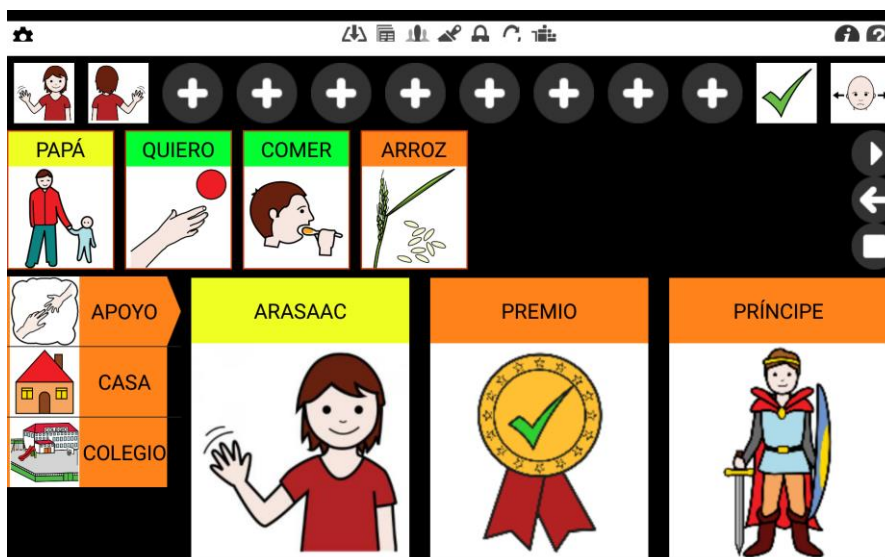


Ilustración 8: CPA, captura 1

Esta aplicación sí que dispone de un modo de enseñar frases en forma secuencial pictograma por pictograma.

LetMeTalk



Ilustración 9: LetMeTalk logo

Esta aplicación sirve para la creación de frases acompañadas de pictogramas. Está más orientada a ser un comunicador que un planificador visual de actividades, pero puede utilizarse también con esta finalidad. Puede ser utilizada tanto para móviles como para tablets.

Utiliza los pictogramas proporcionados por ARASAAC y estos aparecen ordenados en diferentes categorías. También permite la adición de imágenes propias a través de la galería de imágenes del dispositivo. No permite tomar fotos en el acto, primero habría que tomar la foto y añadirla posteriormente.

Permite compartir los pictogramas presentes en una tira, pero lo hace de manera separada. Se exporta y envía cada pictograma como un archivo de imagen independiente. También permite exportar los datos de la aplicación, esto incluye las diferentes categorías de pictogramas con los pictogramas y categorías añadidas o creadas por el usuario.

El diseño de la interfaz es simple, es fácil de utilizar, se hace uso de botones grandes y claros. No requiere de un gran esfuerzo para su aprendizaje. Incluye algunas acciones que no se ven a simple vista como borrar pictogramas de la tira mediante una pulsación larga, esto puede ser confuso para el usuario. La interfaz requiere añadir los pictogramas a un repositorio de pictogramas ordenados por categorías antes de poder utilizarlos, esta decisión hace que el flujo de trabajo sea más lento.

La aplicación incluye un buscador de pictogramas y es la mejor implementación entre las aplicaciones analizadas. Realiza búsqueda filtrada y pregunta al usuario con que palabra asociada añadir el pictograma para trabajar con él.

Por último, la aplicación permite leer con audio la secuencia de pictogramas seleccionada, esta es una de las principales características de la aplicación.

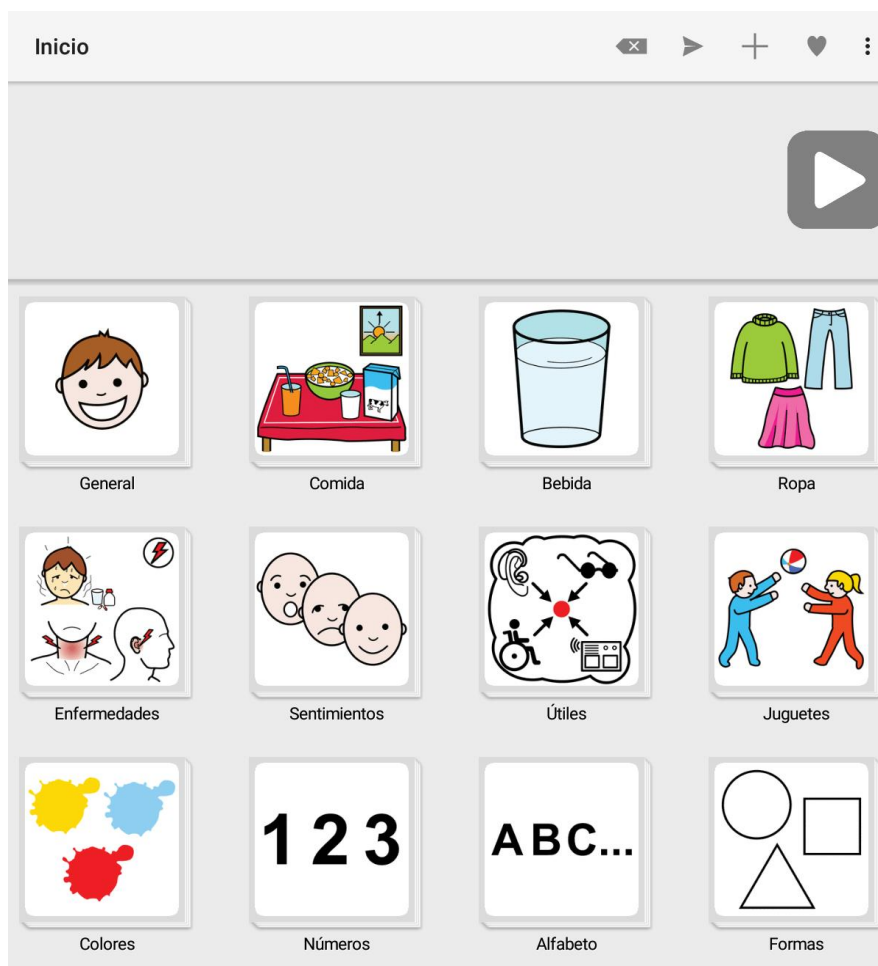






Ilustración 10: LetMeTalk, captura 1

A modo de conclusión de este apartado, se incluye una tabla resumen de características destacadas en las aplicaciones analizadas.

Las filas de la tabla se han elegido mediante una síntesis de características comunes y características deseables de cara a cumplir con los objetivos que se marcaron en el apartado 1.3 (Objetivos) de este mismo documento.

Tabla 1: Comparativa de aplicaciones actuales

				
Funciona en Móvil/Tablet	Sí	Sí	Sí	Sí
Dispone de un tutorial	No	No	Sí	No
Pictogramas ARASAAC	Sí	Sí	Sí	Sí
Imágenes Personales	Sí	Sí	Sí	Sí
Compartir Planificadores	Sí	Sí	No	No
Diseño fácil de usar	Sí	No	No	Sí
Incluye Buscador de Pictos	No	No	Sí	Sí
Incluye un Modo Secuencia	No	Sí	Sí	Sí
Selección de Pictogramas Clave	No	No	No	No
Utiliza sistemas de Comunicación por Colores	Sí	No	Sí	Sí

De esta tabla resumen se pueden extraer diferentes conclusiones, como, por ejemplo, la importancia que tienen las características comunes en todas las apps analizadas.

La aplicación a desarrollar deberá funcionar en tablets y móviles. También, hará uso de los pictogramas de ARASAAC y posibilitará el uso de imágenes y fotos propias.

Llama la atención que no todas las aplicaciones hagan disposición de una forma de compartir los planificadores o los tableros creados en algún formato estándar como pdf, o png/jpeg.

Por otra parte, el diseño de todas las aplicaciones es mejorable y no se tienen en cuenta las necesidades del usuario final. El buscador de pictogramas debe de ser una herramienta imprescindible en este tipo de aplicaciones donde se maneja una gran cantidad de imágenes. La posibilidad de añadir colores a los pictogramas es una característica a tener en cuenta de cara a la aplicación a desarrollar.

2.5. Alternativas para la implementación

De cara a la implementación del sistema se evalúan diferentes alternativas de cara al desarrollo. La tecnología a utilizar para desarrollar el producto final es menos importante que la utilidad del producto en sí, pero, es importante realizar un estudio de las alternativas existentes. Se evalúan principalmente tres alternativas.

2.5.1. Aplicación en código nativo (Objective-C, Java)

Una manera de desarrollar la aplicación sería a través de código nativo para una plataforma determinada. Para Android, se podría elegir programar en Java o Kotlin y para iOS programar en Objective-C o Swift.

La principal ventaja sería el rendimiento, el principal problema es que para hacer una aplicación que funcione en ambas plataformas habría que programar la aplicación 2 veces.

2.5.2. Aplicación Híbrida (Cordova, Ionic)

La siguiente alternativa sería desarrollar una aplicación híbrida. Una aplicación híbrida es una aplicación web clásica compuesta por HTML, CSS y Javascript que está contenida en un WebView. Un WebView funciona tanto en iOS como en Android. El funcionamiento de la aplicación sería igual al funcionamiento de una aplicación nativa, pero por debajo, cordova corre un pequeño servidor en el que se ejecuta la aplicación web.

Las ventajas de usar un framework como, por ejemplo, ionic, es que existen una gran cantidad de componentes ya creados que se pueden utilizar. Por otra parte, la aplicación desarrollada tendría el mismo funcionamiento tanto para Android como para iOS.

La principal desventaja es que el rendimiento es inferior al del código nativo.

2.5.3. Aplicación compilada a código nativo (React native)

Por último, hay una tercera alternativa que es desarrollar con un framework del estilo React Native, que permite desarrollar apps multiplataforma con Javascript. La diferencia con una app híbrida es que el código se compila a código nativo.

Es multiplataforma, pero no al 100%, hay componentes que solo funcionan para una determinada plataforma, y, en cierto modo, hay que programar la aplicación más de una vez. Por otra parte, se trata de un framework relativamente nuevo y aún tiene apartados que mejorar.

3. ANÁLISIS DEL SISTEMA

En este apartado se detallarán todas las capacidades y restricciones de la aplicación tomando como punto de partida el estado del arte ya analizado. Además, dado que este trabajo pretende basarse en la metodología de diseño centrado en el usuario, es primordial realizar un ejercicio de identificación de los usuarios finales para conocer así cuáles son sus capacidades y sus necesidades. El objetivo es desarrollar una aplicación para resolver unas necesidades concretas y que tenga una alta usabilidad de cara a los usuarios finales.

Primero, se partirá de este análisis de los distintos usuarios finales de la aplicación para poder elicitar requisitos más adelante. Todos los requisitos funcionales y no funcionales deberán partir de este análisis para poder garantizar que la aplicación es usable por parte de los usuarios finales.

3.1. Identificación de Usuarios

Los principales grupos de usuarios a los que se dirige la aplicación son los siguientes.

- Educadores en clase de educación especial.
- Padres de niños con deficiencias cognitivas.
- Personas con deficiencias cognitivas capaces de aprender realizar sus propios planificadores con pictogramas.

El principal grupo de usuarios al que se dirige la aplicación son los educadores en clase de educación especial dado que necesitan proporcionar a sus alumnos de estos soportes visuales en clase de manera diaria. Realizar planificadores visuales forma parte de su rutina y, sobre todo, de la rutina diaria de sus alumnos. Los educadores se ven obligados a invertir mucho tiempo en la elaboración de estos planificadores y como ya se ha visto, no hay ninguna aplicación destinada a la elaboración de anticipaciones que funcione bien y que sea usable, por lo que se hace uso de herramientas de propósito general. Más adelante en este mismo apartado se realiza un análisis de las herramientas que usan actualmente los educadores para conocer bien los problemas que afrontan.

Por otra parte, la aplicación se dirige a los padres, padres que no disponen de las capacidades técnicas para llevar a cabo estos planificadores visuales por su cuenta. Las aplicaciones disponibles actualmente no son lo suficientemente fáciles de usar y no ofrecen la ayuda suficiente para llevar a cabo un planificador visual que pueda interpretar de manera satisfactoria un niño con TEA.

De manera adicional, esta aplicación estaría dirigida a cualquier persona con TEA con la capacidad de aprender a desarrollar sus propios planificadores con pictogramas, de este caso se hace mención en el artículo de Takanori Koyama & Hui-Ting Wang (2011).

La principal necesidad de cara a padres y personas con TEA sería realizar una aplicación accesible y fácil de usar.

Por último, no hay que olvidar que el objetivo de este proyecto es el de mejorar la calidad de vida de los niños con TEA en clases de educación especial. A esto se pretende llegar mediante los planificadores visuales creados por sus educadores y por sus padres, pero también son usuarios finales de la aplicación, pues serán los que consuman el contenido que se va a crear con esta aplicación.

3.2. Problemas de los educadores

Del propio estado del arte y de los problemas definidos en el apartado 1.2, se pueden empezar a definir las necesidades y dificultades que tienen los educadores para llevar a cabo su trabajo.

De entre todos los problemas que tienen los educadores, destacan el verse obligados al uso de herramientas de propósito general, lo cual conlleva una pérdida de tiempo y una mala experiencia por los impedimentos propios de estas herramientas.

3.3. Análisis de la metodología actual

Para obtener una visión más personal de la experiencia de usuario con estas herramientas y conocer los impedimentos con los que lidian los educadores, se ha llevado a cabo el ejercicio de realizar un planificador con pictogramas con Microsoft Word y Microsoft PowerPoint.

En este análisis se podrán observar los principales problemas que tienen estas herramientas y de esta forma reunir todos los aspectos en los que mejorar de cara al desarrollo de la aplicación.

El objetivo es generar un documento en formato A4 y orientación horizontal para que posteriormente pueda ser impreso para su utilización en clase. El documento consistirá en una serie de pictogramas que vendrán acompañados de texto. El planificador es una copia de un anticipador real que se ha utilizado en clases.

3.3.1. Microsoft Word

En este punto se detalla el proceso a seguir para la creación de un anticipador con pictogramas básico con Word.

El primer paso es crear un nuevo documento. Se puede elegir entre crear un documento en blanco desde cero o utilizar alguna plantilla disponible. Dado que ninguna de las plantillas que se ofrecen son de utilidad, se procede creando un documento en blanco.

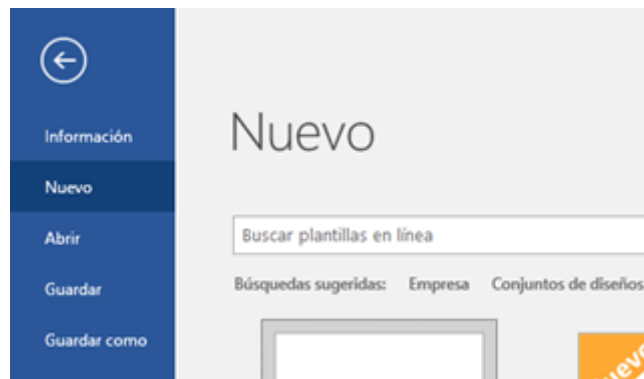


Ilustración 11: Proceso en Word, captura 1

Para rotar el documento creado es necesario acceder a la pestaña Formato. Desde ahí se puede modificar la orientación del documento a horizontal.

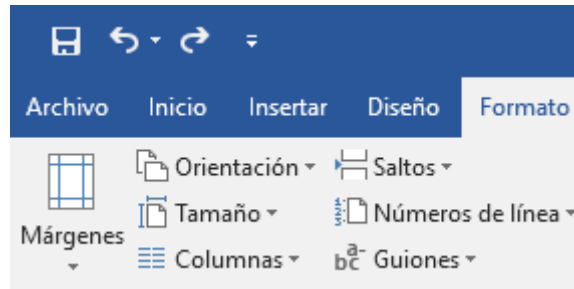


Ilustración 12: Proceso en Word, captura 2

El siguiente paso es insertar los pictogramas deseados y acompañarlos con texto. Para insertar el pictograma se debe hacer uso de la pestaña Insertar, después elegir la opción de insertar imágenes.

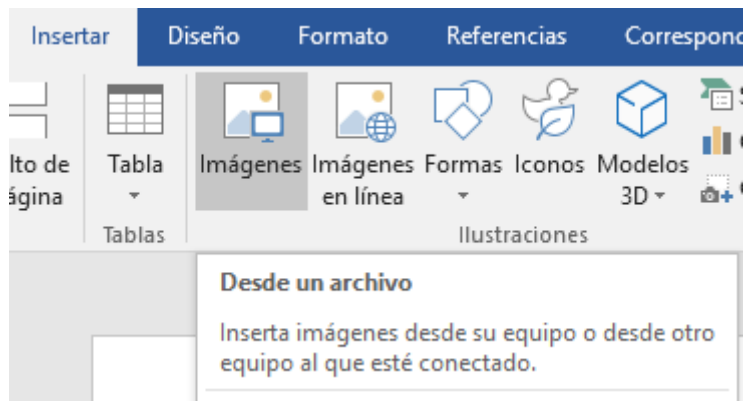


Ilustración 13: Proceso en Word, captura 3

A continuación, hay que buscar y seleccionar el pictograma deseado mediante el explorador en la ventana emergente.



Ilustración 14: Proceso en Word, captura 4

Una vez importada la imagen hay que redimensionarla a un tamaño adecuado. Esto se puede realizar a mano con los controles que rodean la imagen o introduciendo el tamaño específico de la imagen con la herramienta que se puede encontrar en la pestaña “Formato”.

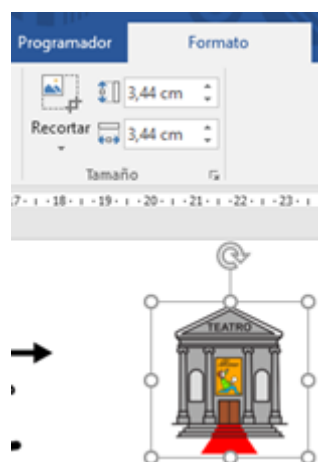


Ilustración 15: Proceso en Word, captura 5

Para mantener la consistencia y la coherencia del planificador, es deseable que todos los pictogramas tengan el mismo tamaño, por lo que habría que introducir a mano valores fijos para el tamaño.

Se procede de la misma manera para insertar todos los pictogramas deseados. Tras insertarlos, es necesario distribuirlos. Para separar los pictogramas lo mejor es usar tabuladores entre ellos.

Para añadir texto debajo de cada pictograma se crea una nueva línea de texto debajo de los pictogramas insertados. De la misma forma, para espaciar el texto se procede mediante tabuladores o espacios. Esta solución no es óptima en términos de alineación con las imágenes, pero es la que lleva menos tiempo en relación a lo que se quiere conseguir.

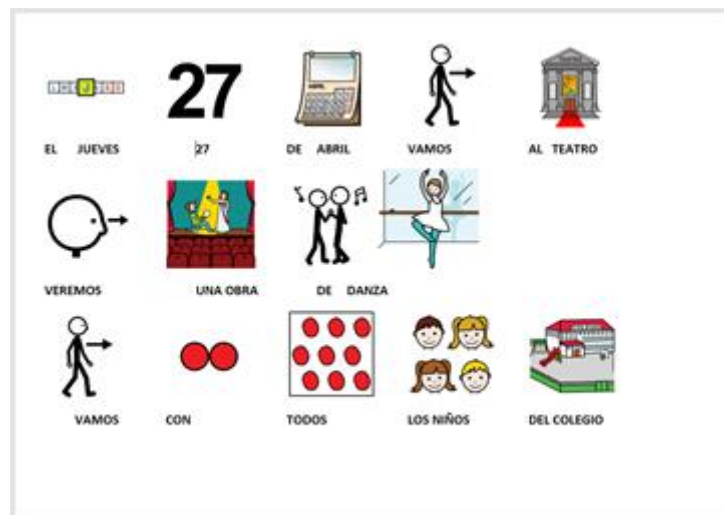


Ilustración 16: Proceso en Word, captura 6

Para una página con la frase “El Jueves 27 de abril vamos al teatro, veremos una obra de danza, vamos con todos los niños del colegio” que contiene 14 imágenes, se ha tardado 20 minutos. Hay que tener en cuenta que un planificador completo se compone de varias páginas, dependiendo de la actividad.

3.3.2. PowerPoint

El proceso para la creación del anticipador con este programa es similar y presenta problemas parecidos, por otra parte, tiene ciertas características que pueden ser útiles.

El primer paso es crear una nueva presentación en blanco. Al igual que antes, las plantillas propuestas no son de gran utilidad para el caso.

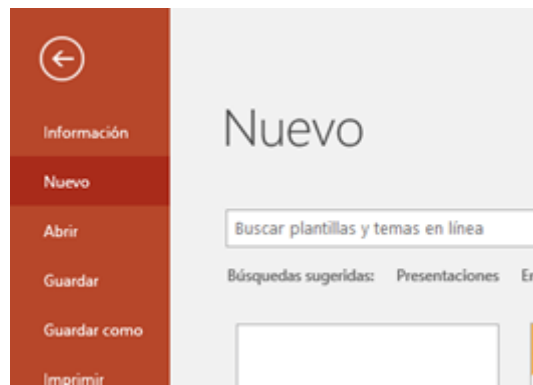


Ilustración 17: Proceso en PowerPoint, captura 1

Hay que modificar el formato de la presentación para que pueda ser impreso en un folio A4. Esto se hace desde la pestaña “Diseño”. Existe una opción para establecer el tamaño deseado de la diapositiva, se debe seleccionar la opción personalizar tamaño de la diapositiva y por último seleccionar A4 con orientación horizontal.

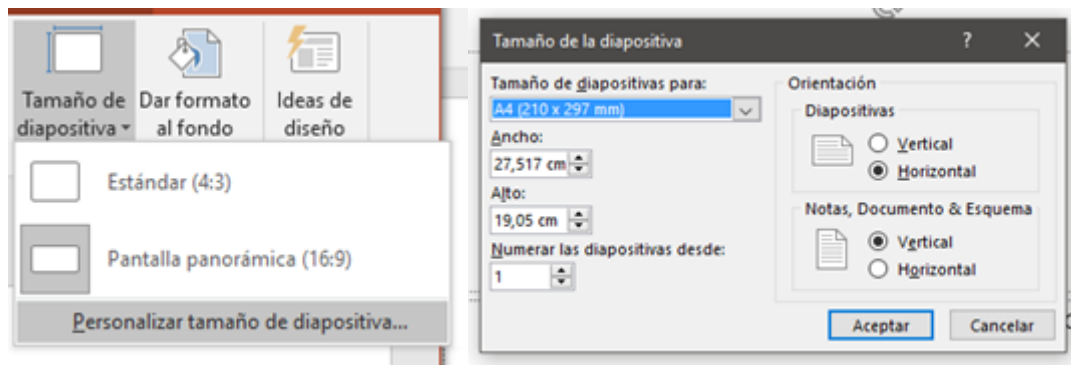


Ilustración 18: Proceso en PowerPoint, captura 2

El proceso de inserción de pictogramas en la diapositiva es el mismo que con MS Word, hay que utilizar la herramienta de inserción de imágenes. Al igual que antes, hay que repetir el proceso de inserción, búsqueda y colocación por cada una de las imágenes a insertar. Esto conlleva navegar por diferentes menús para realizar una acción recurrente.

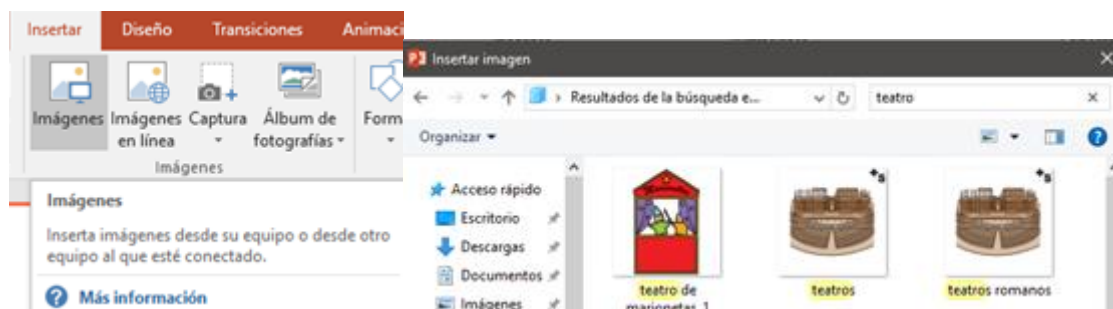


Ilustración 19: Proceso en PowerPoint, captura 3

En este caso, PowerPoint ofrece información sobre la colocación de las imágenes. Ofrece guías sobre el tamaño y la alineación de cada imagen, además de si están o no equiespaciadas, de esta forma no sería necesario repetir el proceso de insertar el tamaño numéricamente de cada imagen a mano.



Ilustración 20: Proceso en PowerPoint, captura 4

Para insertar el texto debajo de cada pictograma se hace uso de cuadros de texto. Se puede proceder creando un cuadro de texto por cada imagen o creando un cuadro de texto por cada línea de pictogramas.

De la misma forma que con Microsoft Word, el tiempo invertido en realizar el planificador ronda los 20 minutos.

Tras este análisis de ambos programas se pueden extraer diferentes conclusiones. Al tratarse de herramientas de índole general, estas no están optimizadas para algo específico como es el trabajo que tienen que realizar los educadores en la elaboración de los planificadores visuales.

- Hay tareas recurrentes que requieren de navegar de manera repetitiva por diferentes menús.
- La búsqueda de pictogramas se basa en el explorador de Windows y este tiene aspectos mejorables como la velocidad de búsqueda.
- La alineación de las imágenes y el ajuste de tamaño es un proceso que puede resultar pesado y consume gran parte del tiempo.
- No existe una forma fácil y rápida de alinear el texto con las imágenes, teniendo que recurrir a tabuladores y espacios en Word.
- Es necesario un PC para poder generar estos planificadores.

La experiencia en general consiste en una serie de impedimentos y restricciones propias del programa con los que hay que lidiar, perdiendo así el enfoque en lo que realmente se está haciendo.

De cara a la aplicación a desarrollar, hay que mejorar el tiempo necesario para agilizar las acciones más recurrentes en la aplicación.

La búsqueda de pictogramas debe de ser rápida, la búsqueda filtrada parece la mejor opción puesto que en muchas ocasiones no será necesario escribir el nombre completo del pictograma a buscar para encontrarlo, o se podrán ver distintos pictogramas dónde elegir para una misma palabra.

Dado que la aplicación estará destinada a la creación de planificadores, no será necesario alinear pictograma y texto puesto que irán centrados por pares pictograma-texto.

Por último, la aplicación a desarrollar debe de ser multiplataforma y así flexibilizar su uso en varios dispositivos.

3.4. Definición del Sistema

En el apartado 1.3 (Objetivos), se definieron unos objetivos generales del proyecto, la finalidad de este apartado es desglosar esos objetivos generales en elementos más pequeños y utilizar el análisis realizado para empezar a dar una definición de las capacidades que debe tener el producto final.

El sistema a desarrollar se trata de una aplicación que funcionará en dispositivos móviles, concretamente smartphones y tablets.

Ahora es necesario definir las características imprescindibles que debe poseer la aplicación.

- Permitir la creación de planificadores visuales con pictogramas.
- Hacer uso del repositorio de pictogramas de ARASAAC.
- Permitir la adición de imágenes propias y fotos como pictogramas.
- Incluir un buscador de pictogramas.
- Procurar de un modo de visualización de los planificadores creados, dónde se pueda ver la secuencia de pictogramas del planificador.
- Posibilitar la compartición de los planificadores creados en varios formatos.
- Permitir que la aplicación pueda usarse en distintos tipos de dispositivos.

Una vez que se ha hecho una definición general del sistema, se procede con una definición detallada de los requisitos del sistema.

3.5. Requisitos

A continuación, se definirán los requisitos funcionales y no funcionales del sistema. Los requisitos funcionales van a indicar capacidades que tiene que poseer el sistema y los requisitos no funcionales, restricciones del sistema. Esta fase es importante de cara a una correcta implementación y verificación del sistema final.

Hay que señalar que, la realización de estos requisitos estará sometida a cambios, puesto que se usará una técnica del diseño centrado en el usuario, el diseño iterativo (Abrams, Maloney-Krichmar & Preece, 2004). Los requisitos cambiarán a partir de la evaluación del usuario final, el feedback proporcionado por el usuario final es crítico para garantizar la usabilidad del sistema final.

Primero, se definirá la estructura que va a tener un requisito, así como la plantilla que se va a usar para definir los requisitos. Un requisito se compondrá de los siguientes campos.

- **Identificador:**
 - Requisito Funcional: RFXX
 - Requisito No Funcional: RNFX

Además, un requisito puede ser (*modificado*). De ser así, esto aparecerá indicado a continuación del identificador. Se trata de una versión actualizada del requisito. Los requisitos que aparezcan con el indicador (*Inicial*) son requisitos de una primera versión que han sido modificados en versiones posteriores.

- **Nombre:** Palabra o frase descriptiva del requisito.
- **Fuente:** Origen del requisito, puede ser del alumno, del tutor o del educador entrevistado.
- **Necesidad:** Imprescindible o Deseable. Dependiendo de si el requisito se puede omitir o no por falta de tiempo o recursos. Los imprescindibles no se pueden omitir.
- **Descripción:** Explicación concisa del requisito.

La plantilla sobre la que se definirán los requisitos es la siguiente.

Tabla 2: Plantilla de Requisitos Software

RF-XX / RNF-XX	
Nombre	
Fuente	
Necesidad	
Descripción	

3.5.1. Requisitos Funcionales

Tabla 3: Requisito Funcional

RF-01 (Inicial)	
Nombre	Crear Planificador
Fuente	Alumno
Necesidad	Imprescindible
Descripción	<p>El sistema debe poder permitir la creación de planificadores con pictogramas y texto. Un planificador se compone de:</p> <ul style="list-style-type: none"> ● ID: Identificador único generado por el sistema. ● Nombre: Viene dado por el usuario y es un campo obligatorio. ● Preview: Imágen miniatura del planificador a modo de vista previa. ● Elementos: Pares de pictogramas/texto, que genera el usuario. Un elemento debe contener al menos un pictograma, o al menos una cadena de texto, pero no puede estar vacío. ● Tipo: Indica la composición del planificador, puede ser de 3 tipos: <ul style="list-style-type: none"> ○ Imágenes y Texto ○ Sólo Imágenes ○ Sólo Texto

Tabla 4: Requisito Funcional 02

RF-02	
Nombre	Pantalla Principal
Fuente	Alumno
Necesidad	Imprescindible
Descripción	<p>El sistema debe contar con una pantalla principal en la que aparecerán todos los planificadores con pictogramas creados.</p> <p>Cada planificador aparecerá en una tarjeta con una miniatura de la preview y el nombre de planificador.</p> <p>Además, el sistema mostrará por cada planificador opciones para eliminar, compartir y editar el planificador.</p> <p>Por último, desde la pantalla principal se podrá acceder a la pantalla de creación de planificadores.</p>

Tabla 5: Requisito Funcional 03

RF-03	
Nombre	Eliminar Planificador
Fuente	Alumno
Necesidad	Imprescindible
Descripción	<p>Un planificador puede ser eliminado del sistema, esta acción se realiza mediante la pantalla principal y con previo aviso de confirmación.</p> <p>Cuando un planificador se elimina, se borrarán todos los datos del sistema.</p>

Tabla 6: Requisito Funcional 04

RF-04	
Nombre	Compartir Planificador
Fuente	Alumno
Necesidad	Deseable
Descripción	<p>Un planificador puede ser compartido mediante la pantalla principal. El planificador debe poder ser compartido en 2 formatos, el usuario podrá elegir entre:</p> <ul style="list-style-type: none"> • PDF: El sistema genera un documento PDF en formato A4 horizontal con el contenido del planificador. Incluye el Nombre del planificador y todos los elementos. • Picto: El sistema genera un archivo JSON con el contenido de un planificador.

Tabla 7: Requisito Funcional 05

RF-05	
Nombre	Editar Planificador
Fuente	Alumno
Necesidad	Imprescindible
Descripción	<p>Cuando el usuario quiere editar un planificador, el sistema lanzará una pantalla de edición con el contenido del planificador a editar.</p> <p>El usuario podrá editar todos los campos del planificador.</p> <p>Cuando el planificador sea editado y los datos se guarden, el sistema regresará a la pantalla principal.</p>

Tabla 8: Requisito Funcional 06

RF-06	
Nombre	Vista de Planificador
Fuente	Alumno
Necesidad	Imprescindible
Descripción	<p>El sistema deberá tener una pantalla donde se podrá visualizar el contenido del planificador seleccionado.</p> <ul style="list-style-type: none"> • Si el planificador es de texto únicamente, solo aparecerá el texto. • Si el planificador es solo de pictogramas, solo aparecerán los pictogramas.

Tabla 9: Requisito Funcional 07

RF-07 (Inicial)	
Nombre	Modo Visualización
Fuente	Alumno
Necesidad	Imprescindible
Descripción	<p>El sistema deberá tener una pantalla donde se podrá visualizar el contenido del planificador seleccionado pictograma por pictograma, de manera individual.</p>

Tabla 10: Requisito Funcional 08

RF-08	
Nombre	Elección del tipo de Planificador
Fuente	Alumno
Necesidad	Imprescindible
Descripción	En la creación de un planificador, el sistema deberá contar con un modo de elegir si el planificador contará con imágenes y texto, solo imágenes o solo texto.

Tabla 11: Requisito Funcional 09

RF-09 (Inicial)	
Nombre	Adición de elementos en un planificador
Fuente	Alumno
Necesidad	Imprescindible
Descripción	<p>En la creación de un planificador, el sistema deberá contar con un botón para añadir un elemento a un planificador.</p> <p>Un elemento se compone de un pictograma y un texto asociado. Inicialmente, el pictograma será una imagen predefinida. El texto tendrá la palabra “texto”.</p> <p>Ninguno de los campos será obligatorio de rellenar.</p>

Tabla 12: Requisito Funcional 10

RF-10 (Inicial)	
Nombre	Eliminar un elemento
Fuente	Alumno
Necesidad	Imprescindible
Descripción	Un elemento podrá ser eliminado en la creación o edición de un planificador. El elemento será eliminado independientemente de su posición en el planificador.

Tabla 13: Requisito Funcional 11

RF-11	
Nombre	Seleccionar un pictograma
Fuente	Profesor
Necesidad	Imprescindible
Descripción	<p>Un pictograma podrá ser cambiado y se lanzará una pantalla con un buscador y tres categorías:</p> <ul style="list-style-type: none"> • Más usados • ARASAAC • Personales <p>Cuando se elige un pictograma este aparecerá en el elemento seleccionado.</p>

Tabla 14: Requisito Funcional 12

RF-12	
Nombre	Pictogramas de ARASAAC
Fuente	Profesor
Necesidad	Imprescindible
Descripción	El sistema debe contar con una categoría de pictogramas que contenga todo el repositorio de pictogramas de ARASAAC.

Tabla 15: Requisito Funcional 13

RF-13	
Nombre	Pictogramas más usados
Fuente	Profesor
Necesidad	Imprescindible
Descripción	El sistema debe contar con una categoría con todos los pictogramas más utilizados. Estos pictogramas aparecerán ordenados en orden descendente por cantidad de veces que se ha usado el pictograma.

Tabla 16: Requisito Funcional 14

RF-14	
Nombre	Pictogramas personales
Fuente	Profesor
Necesidad	Imprescindible
Descripción	El sistema debe contar con una categoría que contendrá pictogramas añadidos por el usuario. Existirá una opción para añadir pictogramas del usuario.

Tabla 17: Requisito Funcional 15

RF-15	
Nombre	Añadir pictogramas del usuario
Fuente	Profesor
Necesidad	Imprescindible
Descripción	<p>El usuario podrá añadir imágenes propias como pictogramas personales. El usuario podrá elegir entre:</p> <ul style="list-style-type: none"> • Seleccionar una imagen de la galería de imágenes del dispositivo • Tomar una foto <p>A continuación, el sistema almacenará la foto y recibirá un nombre a elección del usuario. El nombre es obligatorio y será una cadena de caracteres.</p>

3.5.2. Requisitos Modificados

Con los requisitos especificados se diseña y se desarrolla un primer prototipo. Este prototipo se explica con más detalle en el apartado 4.2. A partir del prototipo, se prepara una reunión con el usuario final para recibir feedback con apartados que modificar y añadir.

Después de la primera reunión el educador, que es uno de los usuarios finales, aparecen nuevos requisitos y se modifican otros. Los detalles respecto a la reunión se encuentran en el apartado 4.4 de este mismo documento.

El diseño centrado en el usuario funciona iterando a partir de un prototipo con la retroalimentación del usuario, esto es precisamente lo que se hace. A partir de las necesidades del usuario y las sugerencias que surgen tras la interacción con el prototipo, se hace una segunda iteración del prototipo. Los requisitos nuevos y los requisitos modificados son los siguientes.

Tabla 18: Requisito Funcional 16

RF-16 (Nuevo)	
Nombre	Plantillas
Fuente	Educador
Necesidad	Imprescindible
Descripción	El sistema deberá permitir la creación de plantillas de pictogramas. Una plantilla contiene los mismos campos que un planificador, pero se almacena de manera distinta. El sistema contará con una lista de todas las plantillas creadas.

Tabla 19: Requisito Funcional 17

RF-17 (Nuevo)	
Nombre	Creación de planificador desde plantilla
Fuente	Educador
Necesidad	Imprescindible
Descripción	El sistema deberá permitir la creación de un planificador a partir de una plantilla almacenada. El planificador a crear contendrá la información almacenada en la plantilla.

Tabla 20: Requisito Funcional 18

RF-18 (Nuevo)	
Nombre	Eliminar plantillas
Fuente	Alumno
Necesidad	Imprescindible
Descripción	<p>Una plantilla puede ser eliminada del sistema, esta acción se realiza mediante la pantalla de plantillas y con previo aviso de confirmación.</p> <p>Cuando una plantilla se elimina, se borrarán todos los datos del sistema.</p>

Tabla 21: Requisito Funcional 19

RF-19 (Nuevo)	
Nombre	Grupos de pictogramas
Fuente	Educador
Necesidad	Imprescindible
Descripción	<p>El sistema deberá permitir la agrupación de pictogramas en la creación de un planificador. El grupo se compone de:</p> <ul style="list-style-type: none"> ● Nombre: Cadena de caracteres, opcional. ● Elementos: Pares de pictogramas y texto. ● Color.

Tabla 22: Requisito Funcional 20

RF-20 (Nuevo)	
Nombre	Color de grupo
Fuente	Educador
Necesidad	Imprescindible
Descripción	El sistema deberá permitir la elección del color de fondo en un grupo de pictogramas.

Tabla 23: Requisito Funcional 21

RF-21 (Nuevo)	
Nombre	Eliminación de grupo
Fuente	Alumno
Necesidad	Imprescindible
Descripción	Un grupo podrá ser eliminado por completo en la creación de un planificador. El grupo debe eliminarse del sistema.

Tabla 24: Requisito Funcional 22

RF-22 (Nuevo)	
Nombre	Destacar un elemento
Fuente	Educador
Necesidad	Imprescindible
Descripción	Un elemento podrá ser destacado por medio de un marcador. El sistema permitirá destacar un elemento por medio del marcador.

Tabla 25: Requisito Funcional 01 Modificado

RF-01 (Modificado)	
Nombre	Crear Planificador
Fuente	Alumno, Educador.
Necesidad	Imprescindible
Descripción	<p>El sistema debe poder permitir la creación de planificadores con pictogramas y texto. Un planificador se compone de:</p> <ul style="list-style-type: none"> ● ID: Identificador único generado por el sistema. ● Nombre: Viene dado por el usuario y es un campo obligatorio. ● Preview: Imagen miniatura del planificador a modo de vista previa. ● Grupos: Agrupaciones de elementos que contienen. <ul style="list-style-type: none"> ○ <i>Nombre:</i> Cadena de caracteres. No es un campo obligatorio. ○ <i>Elementos:</i> Pares de pictogramas/texto, que genera el usuario. Un elemento debe contener al menos un pictograma, o al menos una cadena de texto, pero no puede estar vacío. <ul style="list-style-type: none"> ■ Un elemento puede estar destacado o no. Los elementos destacados aparecerán en el modo visualización. ○ <i>Color:</i> Color de fondo a elegir entre colores predefinidos. Por defecto, el fondo será blanco. ● Tipo: Indica la composición del planificador, puede ser de 3 tipos: <ul style="list-style-type: none"> ○ <i>Imágenes y Texto</i> ○ <i>Sólo Imágenes</i> ○ <i>Sólo Texto</i>

Tabla 26: Requisito Funcional 09 Modificado

RF-09 (Modificado)	
Nombre	Adición de elementos en un planificador
Fuente	Alumno, Educador.
Necesidad	Imprescindible
Descripción	<p>En la creación de un planificador, el sistema deberá contar con un botón para añadir un elemento a un planificador.</p> <p>Un elemento se compone de un pictograma y un texto asociado. Inicialmente, el pictograma será una imagen predefinida. El texto tendrá la palabra “texto”.</p> <p>Ninguno de los campos será obligatorio de rellenar.</p> <p>Además, un elemento podrá ser destacado, si un elemento está destacado aparecerá en el modo visualización. Por defecto, un elemento no aparecerá como destacado.</p>

Tabla 27: Requisito Funcional 07 Modificado

RF-07 (Modificado)	
Nombre	Modo Visualización
Fuente	Alumno, Educador.
Necesidad	Imprescindible
Descripción	<p>El sistema contará con una pantalla donde se mostrarán los pictogramas destacados en un planificador. Además, se permitirá elegir el número de pictogramas que se muestran en un rango de 1 a 5 pictogramas.</p>

3.5.3. Requisitos No Funcionales

Tabla 28: Requisito No Funcional 01

RNF-01	
Nombre	Funcionamiento multiplataforma
Fuente	Alumno
Necesidad	Imprescindible
Descripción	El sistema deberá funcionar en dispositivos Android y dispositivos iOS.

Tabla 29: Requisito No Funcional 02

RNF-02	
Nombre	Versión mínima Android
Fuente	Alumno
Necesidad	Imprescindible
Descripción	Para dispositivos Android, el sistema deberá funcionar para versiones del SO Android superiores a 4.1.1

Tabla 30: Requisito No Funcional 03

RNF-03	
Nombre	Versión mínima iOS
Fuente	Alumno
Necesidad	Imprescindible
Descripción	Para dispositivos iOS, el sistema deberá funcionar para versiones del SO superiores a la versión 8.

Tabla 31: Requisito No Funcional 04

RNF-04	
Nombre	Responsividad
Fuente	Alumno
Necesidad	Imprescindible
Descripción	El sistema será responsivo y todas las funcionalidades se mantendrán independientemente de la resolución del dispositivo.

4. DISEÑO Y JUSTIFICACIÓN DE LA SOLUCIÓN

En este cuarto capítulo se describirán con detalle las decisiones de diseño que se han tomado para la implementación del sistema analizado y definido en el capítulo anterior.

En este capítulo se explican y se desarrollan las funcionalidades planteadas en los requisitos, el primer paso es la construcción de un primer prototipo con la interfaz.

4.1. Prototipo inicial

Este primer prototipo se trata de un mockup de la interfaz de usuario. Se ha partido de los requisitos planteados para ir delineando un primer acercamiento a la interfaz de usuario.

El mockup se ha llevado a cabo con la herramienta de diseño gráfico Adobe Photoshop. A continuación, se adjuntan las capturas de las diferentes pantallas de la aplicación.

El sistema tendrá una serie de elementos comunes, como por ejemplo la barra de estado y la barra de navegación. En estos componentes visuales se mostrará información respecto a la situación del usuario en el sistema, en la pantalla principal aparecerá el logotipo y el nombre de la aplicación. Además, esta barra de navegación incluye elementos de navegación como el botón de volver atrás.

La pantalla principal contiene todos los planificadores creados, para cada planificador, mostrará una imagen con el contenido del planificador, el nombre del planificador y las diferentes opciones que ofrece, compartir, editar o eliminar. Además, se incluye un botón en la esquina inferior derecha con el cual se podrá acceder a la pantalla de creación de planificadores.

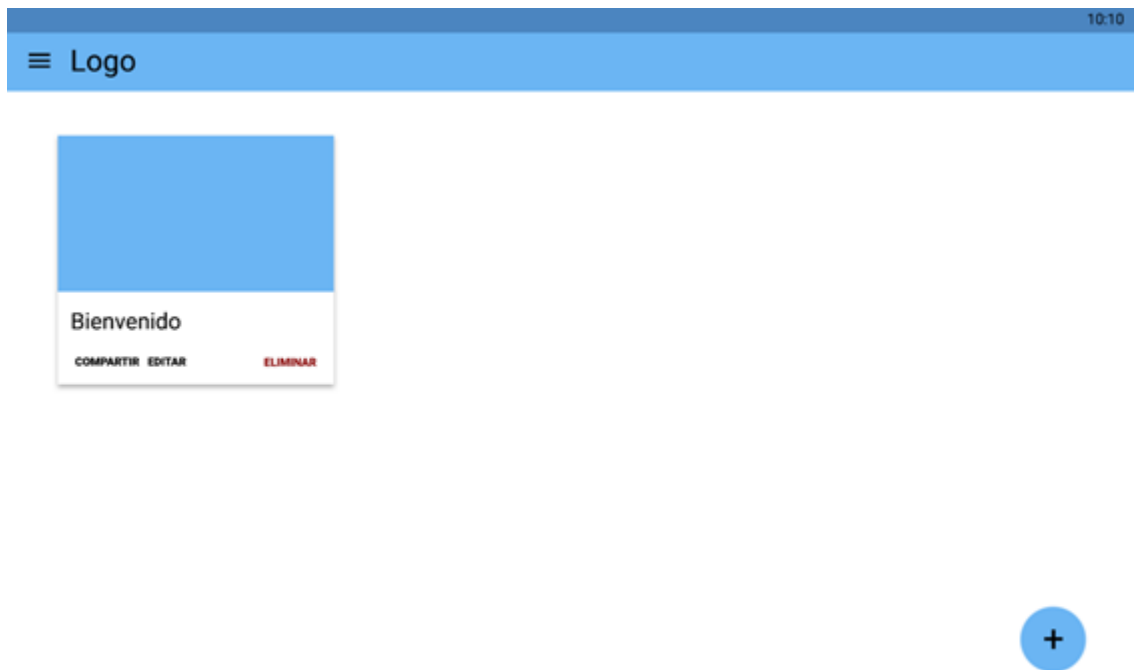


Ilustración 21: Prototipo inicial, Pantalla principal

Para la creación de un planificador, en este mockup inicial se pensó en la creación en dos pasos. En una primera pantalla se establece el nombre del planificador y el tipo de planificador.

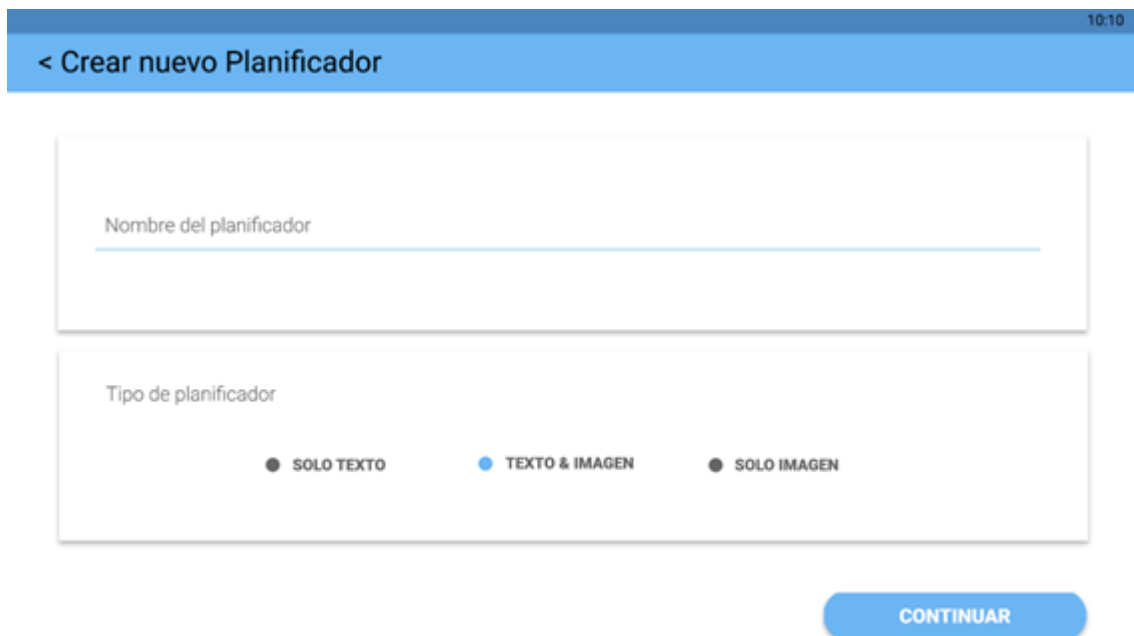


Ilustración 22: Prototipo inicial, Pantalla de creación (I)

Una vez se establecen estos campos, el siguiente paso es el de crear el planificador como tal, añadiendo tantos elementos como fuese necesario. En esta segunda pantalla

aparecerán todos los pictogramas y texto que conforman el planificador, así como un botón con el que añadir más elementos.

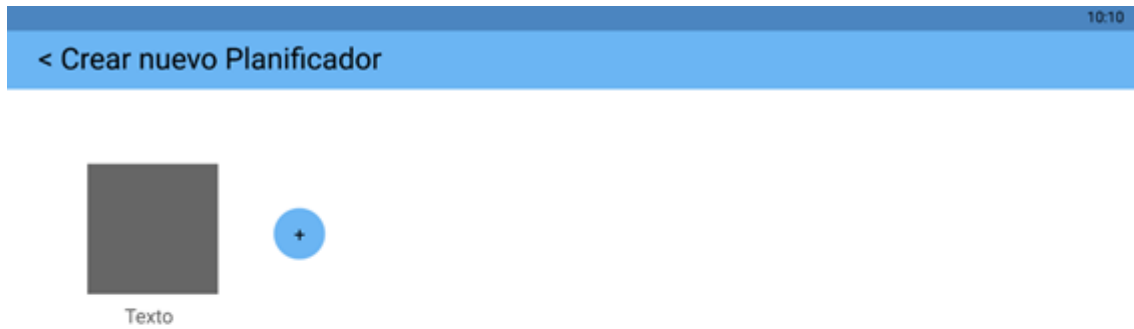


Ilustración 23: Prototipo inicial, Pantalla de creación (II)

Para seleccionar un pictograma, es necesario pulsar en el pictograma que se desea cambiar. Una vez que se ha hecho esto, se lanzará una nueva ventana con todos los pictogramas disponibles ordenados según las 3 categorías definidas. Además, esta pantalla será la que implemente la barra de búsqueda.

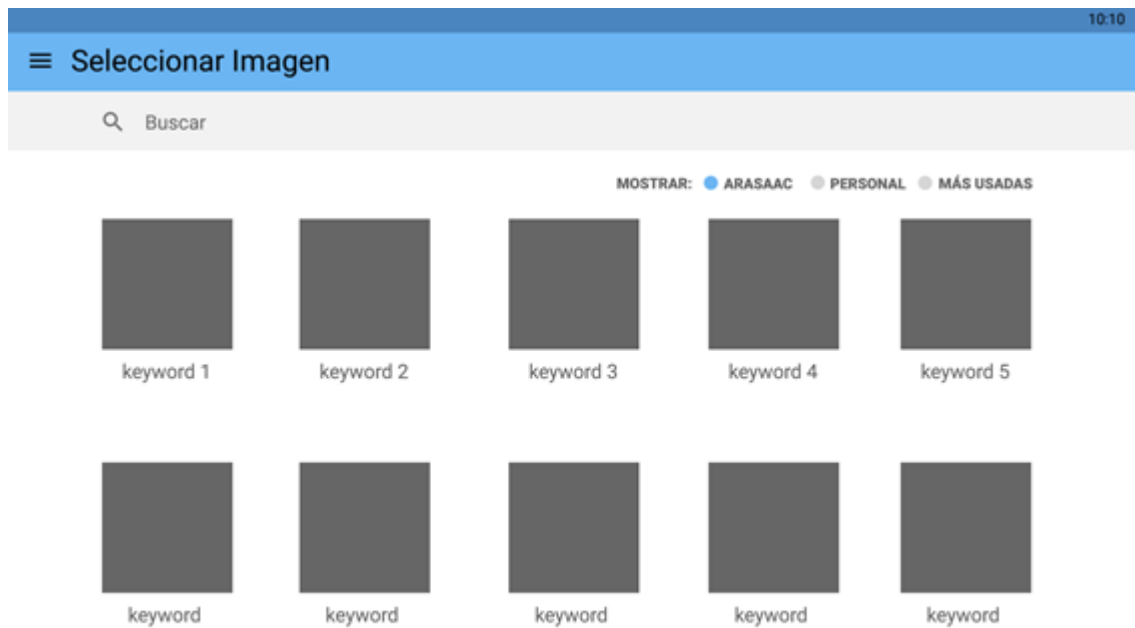


Ilustración 24: Prototipo inicial, Pantalla de búsqueda

Al guardar el planificador, el sistema retorna al usuario a la pantalla principal.

Para visualizar el contenido de un planificador, el usuario debe acceder desde la pantalla principal, seleccionando el planificador a través de una pulsación en la tarjeta del planificador deseado. En la pantalla de visualización, aparecerán todos los elementos del planificador.

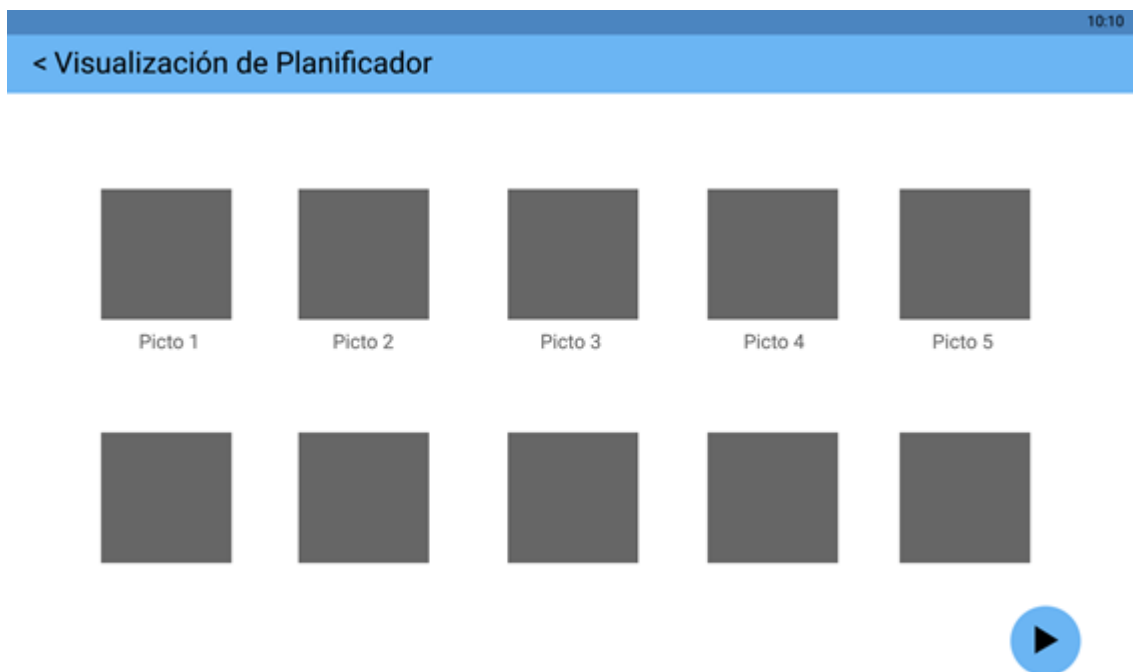


Ilustración 25: Prototipo inicial, Modo de visualización (I)

Pulsando el botón de la esquina inferior derecha se podrá acceder al siguiente modo de visualización. En este modo se muestran todos los elementos del planificador uno a uno. De cara a la utilización con los niños, se debe valorar si es deseable que aparezca el elemento anterior y el elemento posterior.

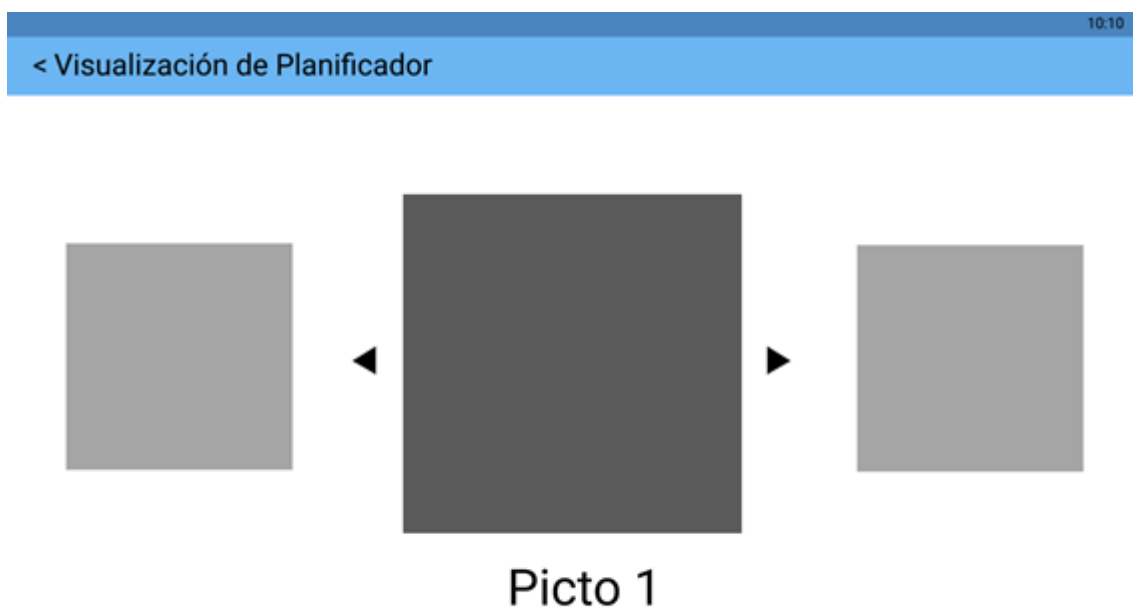


Ilustración 26: Prototipo inicial, Modo de visualización (II)

4.2. Implementación en Ionic Framework

Tras el análisis de las diferentes alternativas para la implementación del sistema, se ha terminado eligiendo Ionic Framework en su versión 3. Este Framework permitirá el desarrollo de una única aplicación que será funcional en dispositivos con diferentes sistemas operativos. Además, los componentes visuales que ofrece este framework son de gran utilidad para el desarrollo del sistema.

Si bien en el apartado 2.5 se menciona que el rendimiento no es el mejor entre las alternativas disponibles, esto no quiere decir que el rendimiento sea malo. La aplicación que se va a desarrollar se trata de una aplicación sencilla y el rendimiento que ofrece ionic es más que suficiente y no representará un problema.

Para la implementación del sistema se seguirá el patrón de arquitectura software Modelo-Vista-Controlador (MVC).

- **Modelo:** Es la representación de la información que maneja el sistema, se encarga de los datos. Realiza todas las operaciones relacionadas con la base de datos, actualizaciones, consultas, búsquedas, etc. Envía datos a la vista a través de peticiones por parte del controlador. En Ionic, el modelo estaría compuesto por todas las clases TypeScript con el modelo de datos y los providers encargados de gestionar las acciones sobre la base de datos.
- **Vista:** Es la representación visual de los datos del sistema, es decir, del modelo. Presenta los datos en una interfaz gráfica lista para interactuar con el usuario. La vista se compone de todas las páginas HTML y SCSS que forman la aplicación.
- **Controlador:** Responde a eventos del usuario realizando peticiones al modelo y actualizando la vista asociada. Se trata de un intermediario entre modelo y vista. En Ionic, el controlador se compone de todas las clases TypeScript con la lógica de la aplicación.

4.2.1. Modelo

Se compone por las clases del modelo de datos, para el prototipo inicial se definen dos clases.

- **Clase Planificador:** Contiene los atributos definidos en los requisitos.
 - **ID:** Identificador único de tipo String
 - **Name:** Nombre para el planificador de tipo String
 - **Type:** String que indica el tipo de planificador (texto, imágenes, ambos)

- Items: Array de elementos de clase Item
- Preview: URL a la imagen miniatura del planificador
- Clase Item: Contiene los pares imagen/texto.
 - Image: URL a la imagen del elemento
 - Text: Texto asociado al elemento

Además, para guardar los datos se hace uso del plugin “ionic-storage”, este plugin hace uso de la base de datos SQLite para el almacenamiento de datos en Android y iOS. El funcionamiento del plugin es simple, funciona almacenando pares de clave/valor.

A partir de ionic-storage se define un “data-provider” en el que se definen métodos para el manejo de datos. Estos métodos son de guardado, recuperación, actualización y borrado.

Todos estos métodos serán invocados por todas las clases TypeScript que forman el controlador.

4.2.2. Vista

La vista de la aplicación se compone de todas las páginas definidas para el sistema. Para el prototipo inicial, las páginas presentes en el sistema son.

- Página Principal
- Página de Edición/Creación
- Página de Selección de Pictograma
- Página de Visualización del planificador
- Página de Visualización de pictogramas

Para la implementación de la interfaz siguiendo el mockup planteado se ha hecho uso de los componentes que ofrece ionic. Cabe destacar que ionic utiliza muchas de las guías de diseño planteadas por Google y su Material Design (Google, 2018).

4.2.3. Controlador

Según la forma en la que se estructura el framework de ionic, cada página tiene su propio archivo TypeScript con la lógica necesaria para el funcionamiento del sistema.

Para cada página, primero se importan todos los módulos necesarios, estos son tanto módulos nativos de ionic como módulos de terceros. A continuación, se definen variables globales y se declara el constructor de la clase.

Además, cada clase está compuesta por una serie de métodos que se ejecutarán con la llegada de eventos al sistema, es decir, acciones realizadas por el usuario. Estos métodos realizarán cambios sobre el modelo y actualizarán la vista.

4.3. Prototipo en ionic

A continuación, se adjuntan capturas del prototipo desarrollado en ionic explicando las diferentes decisiones de diseño de cada página.

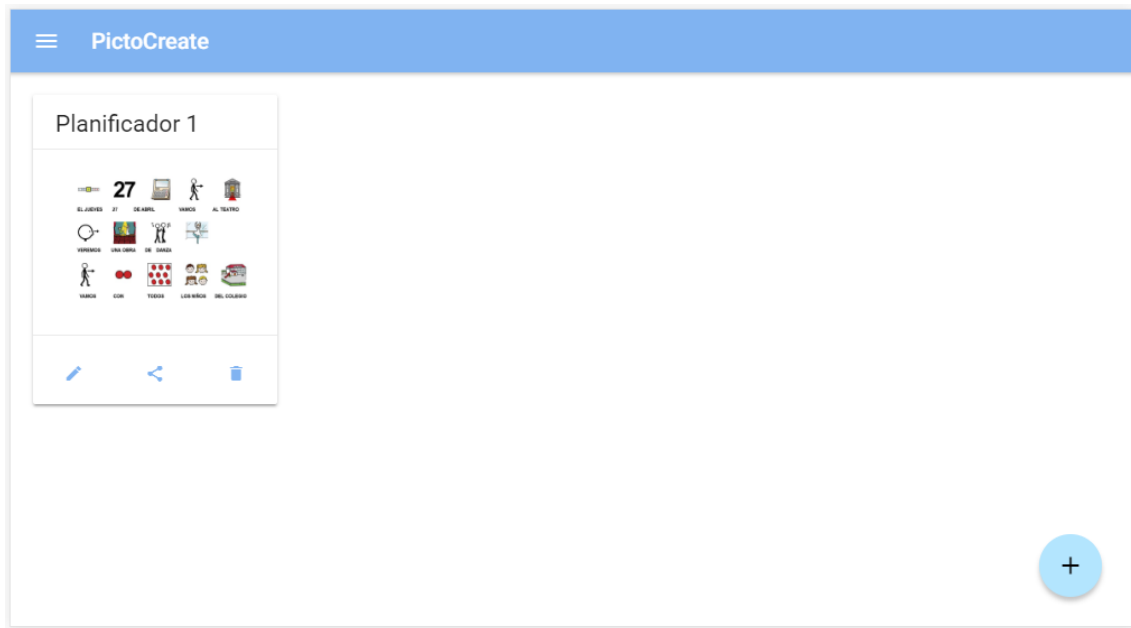


Ilustración 27: Prototipo en ionic. Pantalla Principal

En esta pantalla se pueden ver los planificadores creados como “ion-cards”. La tarjeta cuenta con botones para realizar las acciones de editar, compartir y eliminar. Los iconos seleccionados pretenden ser representativos de las acciones que realizan.

El botón para crear un planificador es un Floating Action Button (FAB), este tipo de botón está presente en otras aplicaciones populares como Twitter o Whatsapp y, por lo tanto, se espera que el usuario intuya su funcionalidad.

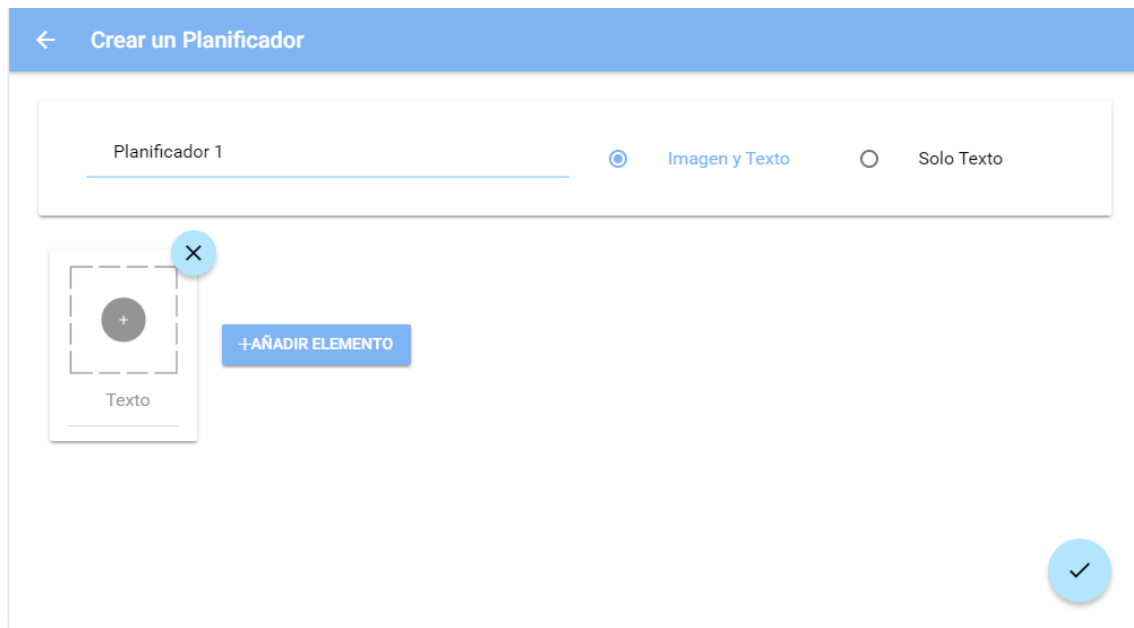


Ilustración 28: Prototipo en ionic. Creación de un planificador

Esta pantalla es la de creación o edición de planificadores. En la barra de navegación se puede leer el estado del sistema “Crear un Planificador” o “Editar un Planificador”, de esta forma el usuario puede saber su localización en el sistema y el estado del mismo.

En la zona superior izquierda se encuentra el campo donde escribir el nombre del planificador y en la zona superior derecha se puede seleccionar el tipo de planificador.

El resto de la pantalla queda libre para la creación del contenido del planificador. Los elementos aparecen en tarjetas. Con la imagen que se ha elegido por defecto se pretende que el usuario intuya que debe pulsar sobre la tarjeta para realizar la acción de seleccionar el pictograma. En la parte inferior de la tarjeta se encuentra un campo para introducir el texto correspondiente al elemento y en la parte superior derecha aparece un botón para eliminar el elemento.

A continuación del último elemento añadido, aparece un botón para seguir añadiendo elementos “+ Añadir elemento”. Siguiendo la ley de Fitts, el botón se encuentra inmediatamente al lado del último elemento añadido, aprovechando la cercanía espacial con la última acción, además, se trata de un botón grande en el eje X, que es el eje de la acción.

Por último, en la parte inferior derecha se encuentra el FAB para guardar los cambios y crear el planificador.



Ilustración 29: Prototipo en ionic, Selección de pictogramas

Para seleccionar los pictogramas, se dispone de las tres categorías definidas como pestañas en la zona superior. En esta misma zona superior se encuentra la barra de búsqueda. Cuando se escribe en la barra de búsqueda los pictogramas se filtran, quedando únicamente aquellos relacionados con la palabra escrita.

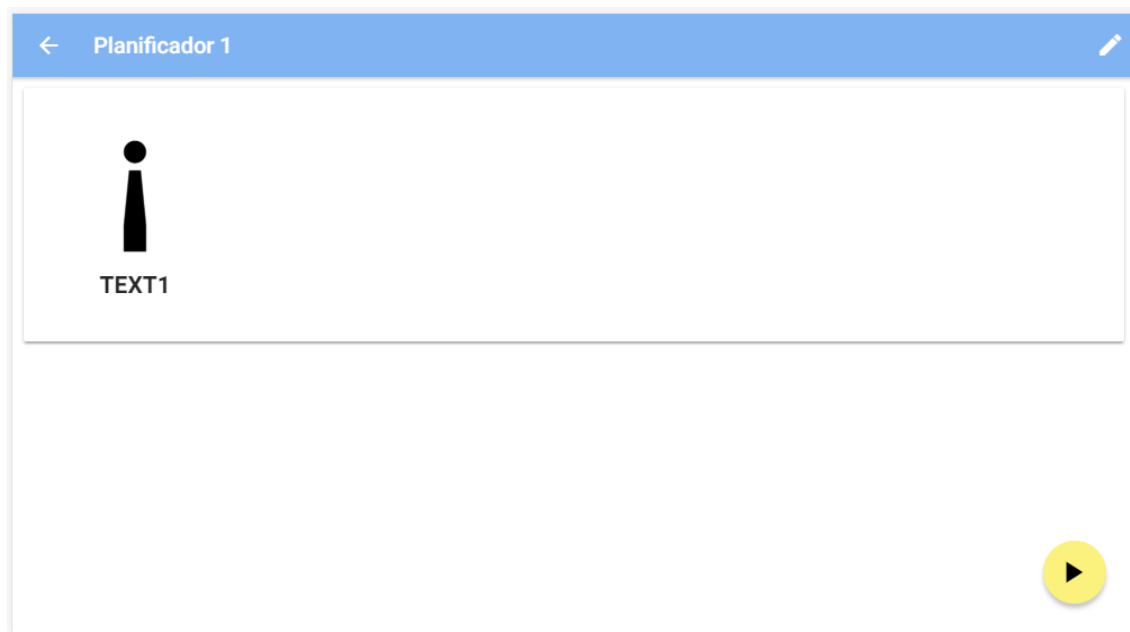


Ilustración 30: Prototipo en ionic, Modo de visualización del planificador

En la pantalla de visualización aparecen todos los pictogramas del planificador. En la barra superior aparece el nombre del planificador. Para acceder al siguiente modo de visualización es necesario pulsar el FAB en la esquina inferior derecha.

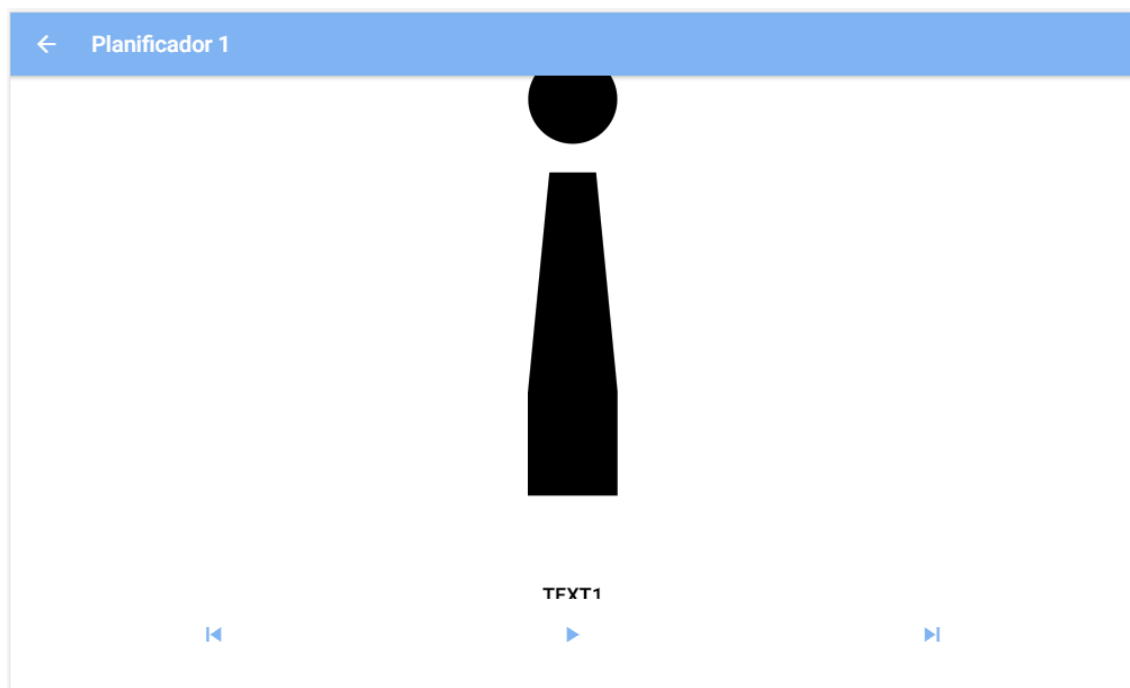


Ilustración 31: Modo de visualización de pictogramas

En este último modo de visualización se podrán ver todos los elementos del planificador uno por uno. Deslizando se puede acceder al siguiente o al anterior.

4.4. Reunión con el educador

Con este primer prototipo funcional desarrollado en ionic se prepara una reunión con un educador de un centro de educación especial. La reunión consiste en una entrevista y en una fase de testeo de la aplicación en la cual el educador puede dar su retroalimentación respecto a apartados que funcionan, apartados donde mejorar, posibles adiciones, etc.

4.4.1. Entrevista

Con el objetivo de obtener datos para el desarrollo de la aplicación, se prepara una entrevista. La entrevista tiene una duración de aproximadamente media hora y se elabora un guion con una serie de preguntas para tener una estructura que seguir. Los datos del educador se mantendrán anónimos de acuerdo al consentimiento informado firmado. Para llevar a cabo la entrevista es necesario que el entrevistado de su consentimiento informado, este se puede ver en el Anexo I.

A continuación, se hace un resumen de los puntos más importantes de la entrevista.

Para empezar, se hacen preguntas para conocer el conocimiento tecnológico del educador. Cuántos años lleva en la enseñanza y si está habituado a la utilización de dispositivos tecnológicos.

El entrevistado lleva 11 años en la enseñanza formal y es el coordinador de tecnologías de la información y comunicación del centro de enseñanza. Es el encargado de poner en marcha todos los dispositivos tecnológicos que llegan al centro. En clases disponen de una pizarra electrónica y tablets.

Para comprender el alcance de la aplicación desarrollada se pregunta cómo de común es el uso de planificadores visuales durante clases.

“Los planificadores visuales son un soporte que se utiliza todo el día y todos los días. La diferencia entre acudir a una actividad con un planificador visual anticipador o sin él es muy notoria. Los niños están muy agitados, salen corriendo, se tiran al suelo, piden constantemente comida, es imposible. Con un anticipador los niños son capaces de acudir a las actividades anticipadas con mucha tranquilidad.”

Respecto a aplicaciones existentes, el educador dice:

“Existen aplicaciones de pictogramas, comunicadores con pictogramas, agendas de comunicación, pero no me gustan, no son prácticas para la manera en la que yo pienso que los niños deben llevar su planificación porque son fácilmente confundibles para ellos.”

Para comprender mejor el trabajo del educador, se le preguntó por la metodología actual que tiene de desarrollar los anticipadores con pictogramas.

Los planificadores visuales, que para actividades futuras se denominan anticipaciones o anticipadores, los realiza en Microsoft Word.

El proceso que se sigue para los planificadores visuales es el analizado en el punto 3.2.1 de este mismo documento. Para el día de la actividad se elaboran unos acordeones de pictogramas.

Por otra parte, para el día de la actividad se elaboran unos acordeones de pictogramas llamados “planificadores simples” con los pictogramas clave. Este proceso se lleva a cabo a mano. Por medio de la aplicación PictoSelector, se imprimen hojas con los pictogramas clave necesarios. A partir de esto se recortan los pictogramas de la hoja y se pegan en un acordeón de cartulina.

La entrevista termina con preguntas respecto a los formatos para exportar los planificadores. Es de preferencia del entrevistado el formato pdf con el contenido del

planificador por tiras de pictogramas, obteniendo así un producto que es fácil de compartir y de visualizar en diferentes dispositivos.

4.4.2. Prueba de la aplicación

Durante esta reunión, se realizan también pruebas con la aplicación. El educador comenta ciertos puntos a tener en cuenta.

“Los colores de los fondos donde se presenta la información de los pictogramas es importante, se pueden definir diferentes paneles con un significado distinto, por ejemplo, PECS.”

De cara a la elaboración de los planificadores, el educador comenta que los anticipadores suelen tener una estructura concreta, repetir esta estructura siguiendo patrones comunes en la elaboración de planificadores es de gran ayuda para los niños. Una adición interesante sería la posibilidad de añadir agrupaciones de pictogramas donde poder definir una temática similar, por ejemplo, cuándo, cómo, con quién y qué se hará durante la actividad.

Relacionado con esto mismo, se plantea también la posibilidad de incluir plantillas. Si los planificadores tienen una estructura similar, es interesante poder crear un planificador a partir de una plantilla con la estructura predefinida, de esta forma se consigue seguir un patrón definido y ahorrar tiempo de trabajo. En adición a esto, de cara a los padres, la posibilidad de tener plantillas sirve como guía para la realización de planificadores, por ejemplo, un educador puede compartir sus plantillas con los padres para que estos desarrollen sus propios planificadores a partir de una estructura que el educador sabe que funciona.

Por último, de cara al modo de visualización de pictogramas de un determinado planificador, el educador expone que, o siempre se hace uso de todos los pictogramas de planificador, si no solo de los pictogramas más representativos, por lo tanto, se propone el uso de marcadores de pictogramas. De esta forma, en el modo de visualización aparecerán únicamente los elementos que el usuario destaque.

A partir de esta reunión se vuelve a iterar, volviendo a especificación de requisitos software. Los nuevos requisitos aparecen en el apartado 3.5.2 Requisitos Modificados.

4.5. Segundo prototipo

De cara a la implementación del segundo prototipo, es necesario realizar cambios en diferentes pantallas.

Con la adición de las plantillas, se modifica el acceso a la creación de un nuevo planificador. Ahora, con el botón de crear un planificador se accede a una ventana dónde se podrá:

- Crear un planificador nuevo, desde cero.
- Crear un planificador a partir de una plantilla.
- Crear una plantilla.
- Eliminar o compartir una plantilla.

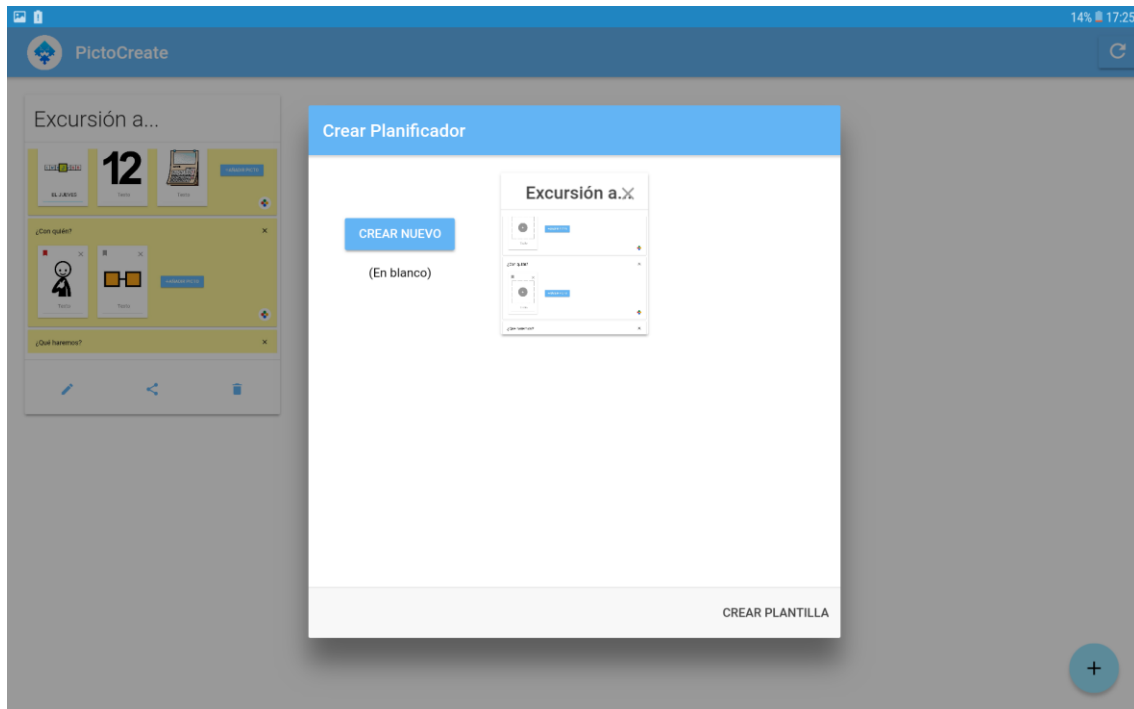


Ilustración 32: Prototipo en ionic (II). Ventana de creación

Para crear una plantilla se sigue el mismo proceso que el de crear un planificador. Como siempre, en la barra de navegación aparecerá el estado del sistema, en este caso “Crear Plantilla”. Una vez creada, la plantilla aparecerá en la ventana de plantillas. En esta captura también se puede observar cómo aparecen las agrupaciones de pictogramas y un botón para seleccionar el color de la agrupación. Para añadir un grupo aparece un nuevo botón debajo de la última agrupación añadida.

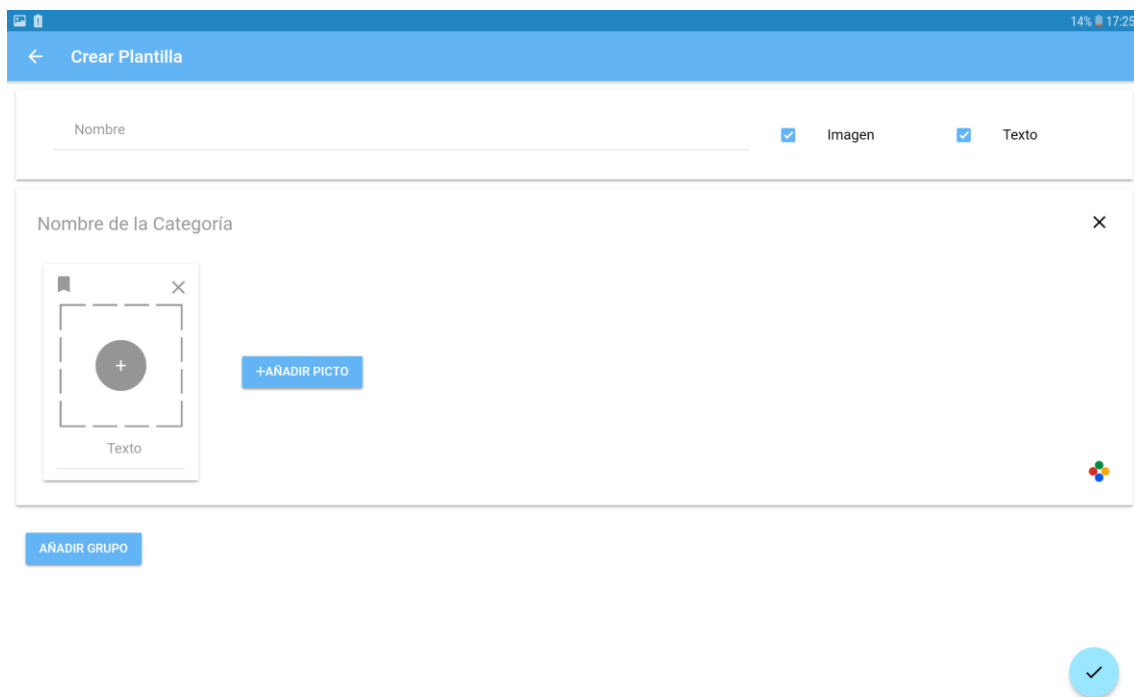


Ilustración 33: Prototipo en ionic (II). Creación de plantilla

Cuando se crea un planificador desde una plantilla, aparecerá el contenido de la plantilla en el planificador a crear. En este ejemplo, se pueden observar varias agrupaciones ya incluidas en la plantilla. Además, también se puede observar el marcador en la esquina superior izquierda de un elemento. Cuando se pulsa el marcador, se vuelve rojo. Un elemento marcado en rojo aparece después en el modo visualización de pictogramas.

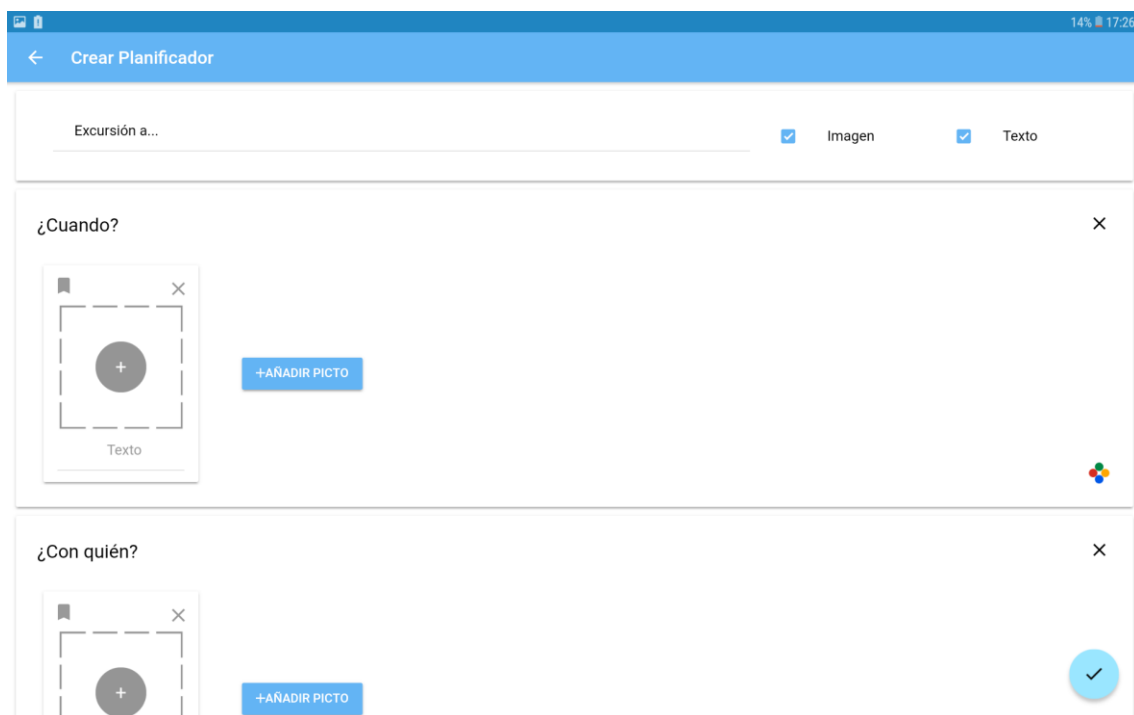


Ilustración 34: Prototipo en ionic (II). Creación desde plantilla

Pulsando sobre el botón en la esquina inferior derecha de una agrupación, se despliega un pequeño menú con los colores disponibles para la agrupación.

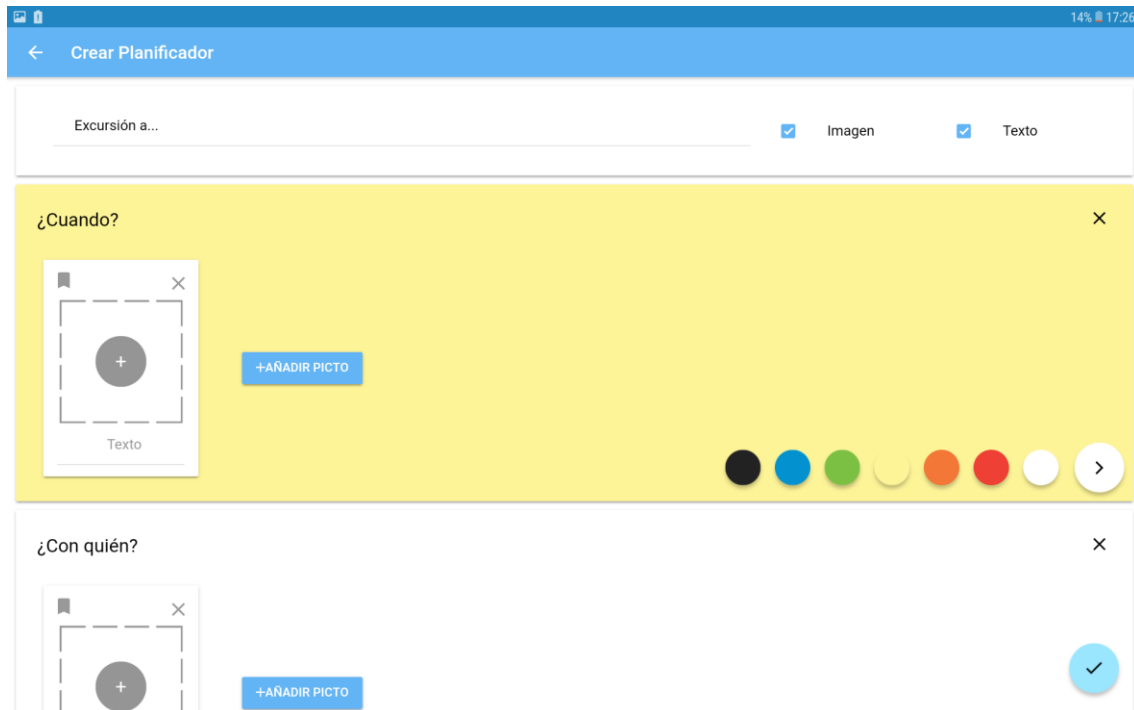


Ilustración 35: Prototipo en ionic (II). Colores de un grupo

En el modo de visualización de pictogramas de un planificador, aparecen todos los pictogramas destacados con un marcador del planificador. En este segundo prototipo se puede elegir el número de pictogramas que aparecen en pantalla entre 1 y 3 pictogramas.

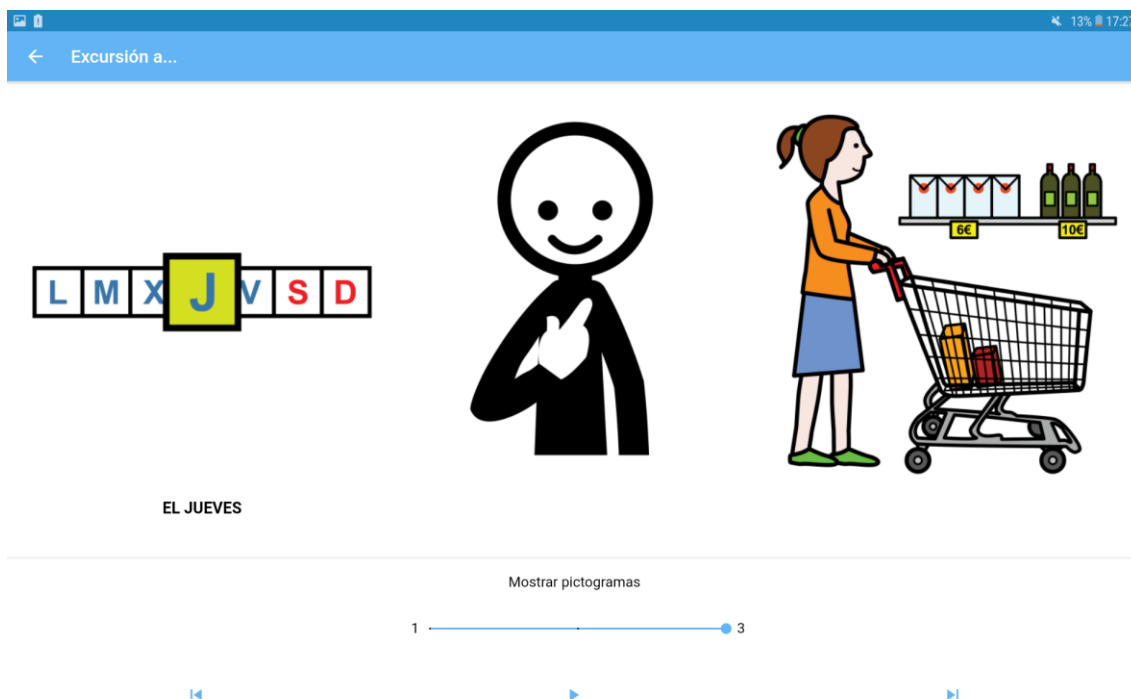


Ilustración 36: Prototipo en ionic (II). Visualización de pictogramas modificado

4.6. Pruebas y trabajo futuro

Con esta versión más avanzada de la aplicación, se realiza una visita a un centro educativo del educador entrevistado en la primera reunión. El objetivo es probar la aplicación en un entorno real y dejar la aplicación para que la utilicen los educadores durante un tiempo. Tras esto, se recogerán otra vez los comentarios y sugerencias de los educadores. El objetivo es realizar al menos una iteración más para obtener un producto más completo y más usable de cara al usuario final.

Esta versión del prototipo es una versión final en vistas a lo propuesto para este trabajo de fin de grado. Durante la reunión se tratan ciertos aspectos de cara a una versión posterior. Entre otras cosas, se plantea como trabajo futuro el desarrollo de un entorno colaborativo entre padres y educadores. De esta forma se pone a disposición de toda una vía directa de colaboración entre padres y tutores en las que, los padres pueden solicitar ayuda de cara a la elaboración de sus propios planificadores y los educadores pueden proveer de esta ayuda a los padres de una manera más rápida y sencilla. Sin embargo, debido a la envergadura de este TFG, esta idea quedará será la semilla para el trabajo futuro.

5. MARCO REGULADOR

En el presente apartado se van a tratar aquellas normas y leyes aplicables al desarrollo de este trabajo. Se realizará un análisis de la legislación aplicable a la implementación del prototipo de la aplicación, así como aspectos como las responsabilidades éticas que se han tenido en cuenta a lo largo de todo el proyecto. Así mismo, se tratarán cuestiones relacionadas con la propiedad intelectual por los contenidos de la aplicación.

5.1. Legislación Aplicable

En el funcionamiento de la aplicación, el usuario lleva a cabo un registro de la información personal respecto a actividades futuras de los usuarios de la aplicación. Esta información puede servir para realizar un ataque de trazabilidad de usuarios prediciendo la localización en un momento determinado de los usuarios.

Debido a esto, es importante conocer la legislación vigente respecto al tratamiento de este tipo de información. Hasta el momento, y durante el desarrollo de la aplicación, se aplicaba la Ley Orgánica de Protección de Datos (LOPD).

Respecto a la protección y tratamiento de datos en la aplicación, cabe destacar que todos los datos recogidos son aportados o facilitados por el usuario bajo su propia responsabilidad y decisión, no existiendo en la aplicación un formulario concreto de recogida de información.

De cara al manejo de los datos, estos son almacenados de forma local en el dispositivo y nunca se utilizan ni se manejan para otros fines distintos a los que el usuario espera.

Sobre la aplicación de la normativa, es importante que se informe al usuario de la finalidad de los datos recogidos y de sus derechos ARCO (acceso, rectificación, cancelación y oposición), que garantizan a las personas el poder de control sobre sus datos personales.

De cara al futuro, en base a la aplicación del Reglamento General de Protección de Datos (RGPD), sería necesario volver a realizar un análisis de riesgos con el objetivo de comprobar si esta nueva normativa se cumple.

5.2. Responsabilidades Éticas

A lo largo de todo el trabajo, se ha hecho uso de un lenguaje denominado Person First Language (PLF), es decir, un léxico y una terminología que pone a las personas primero. Dado que el trabajo gira en torno a personas con distintos tipos de discapacidades cognitivas es necesario conocer e impulsar este tipo de lenguaje.

El lenguaje PLF es una forma objetiva y respetuosa de hablar acerca de las personas con discapacidades en la que se hace énfasis en la persona primero y no en su discapacidad.

La utilización de adjetivos calificativos inexactos o inapropiados genera y promueve estereotipos negativos que devalúan a las personas como seres humanos.

Es importante promover la sensibilización hacia este tipo de lenguaje y recordar que hay ciertas responsabilidades éticas que van más allá de lo tecnológico.

Para conocer más este tipo de lenguaje hay un artículo que se ha tenido en cuenta de Snow, K. (2009).

5.3. Propiedad Intelectual

Para la realización de este trabajo se ha hecho uso del repositorio de pictogramas ofrecido por ARASAAC, por lo tanto, es necesario que este trabajo respete las condiciones de uso establecidas.

Los pictogramas están inscritos en el Registro General de la Propiedad Intelectual conforme al texto refundido de la Ley de Propiedad Intelectual a nombre de la Diputación General de Aragón, por lo tanto, son propiedad del Gobierno de Aragón.

Además, el repositorio de pictogramas ha sido creado por Sergio Palao para ARASAAC y están distribuidos bajo licencia Creative Commons (BY-NC-SA).



Esto implica que, al hacer uso de estos recursos, es necesario que este TFG, como obra derivada, se distribuya con la misma licencia Creative Commons.

6. PLANIFICACIÓN DEL PROYECTO

6.1. Definición de Actividades

La idea del Trabajo de Fin de Grado empieza tras la propuesta del tutor en septiembre de 2017. De cara a elaborar el proyecto, es de vital importancia una buena planificación de las distintas fases que lo conforman para asegurar la consecución de los principales objetivos del trabajo.

6.1.1. Planteamiento

La fase de planteamiento da comienzo al proyecto, en esta fase hay reuniones entre el alumno y el tutor para sentar las bases del proyecto. Se compone de la definición de:

- Contexto
- Problemas
- Objetivos

Esta fase también incluye la parte de investigación de la literatura recomendada por el profesor. Esta actividad se corresponde a la definición del estado del arte de este mismo documento.

Además, también se incluye en este planteamiento el análisis del sistema para el primer prototipo. Esto es, la definición de los requisitos software a partir de un análisis e identificación de los usuarios finales. Dado que se pretende iterar sobre un prototipo, esta fase se repetirá de cara a la elaboración del segundo prototipo.

6.1.2. Desarrollo

La fase de desarrollo se corresponde con todas las actividades relacionadas con la implementación de la aplicación en código. Las actividades se dividen por los diferentes prototipos que se van a llevar a cabo durante el proyecto. En este caso, dos prototipos.

6.1.3. Documentación

Por último, se incluye también una fase de documentación que durará todo el proyecto. Esta fase incluye todas las actividades relacionadas con la elaboración del presente informe. Esta fase también incluye la revisión que se realizará sobre este documento.

6.2. Lista de actividades

Todas las actividades definidas en el apartado anterior se listan a continuación.

1. Planteamiento

- 1.1. Introducción, definición
 - 1.1.1. Contexto
 - 1.1.2. Problemas
 - 1.1.3. Objetivos
- 1.2. Estado del Arte, investigación
 - 1.2.1. TEA
 - 1.2.2. Productos similares
 - 1.2.3. Planificadores Visuales
 - 1.2.4. Tecnología
- 1.3. Análisis del Sistema, prototipo inicial
 - 1.3.1. Identificación de Usuarios
 - 1.3.2. Requisitos

2. Desarrollo

- 2.1. Desarrollo del primer prototipo
 - 2.1.1. Entrevista 1
- 2.2. Desarrollo del segundo prototipo
 - 2.2.1. Entrevista 2

3. Documentación

- 3.1. Escritura
 - 3.1.1. Introducción
 - 3.1.2. Estado del Arte
 - 3.1.3. Análisis del Sistema
 - 3.1.4. Diseño y Justificación
 - 3.1.5. Planificación
 - 3.1.6. Marco Regulador
 - 3.1.7. Entorno Socioeconómico
 - 3.1.8. Conclusiones
- 3.2. Revisión

6.3. Diagrama de GANT

A continuación, se incluye el diagrama de GANTT resultante de la planificación del proyecto por semanas con las fases previamente enumeradas.

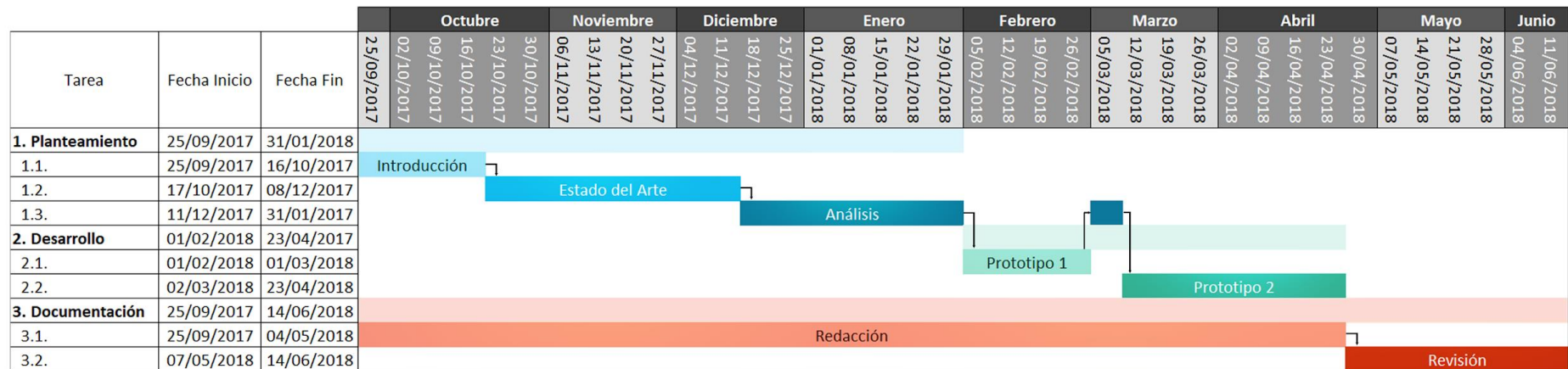


Ilustración 37: Diagrama de GANTT

Durante los primeros cuatro meses del trabajo, las fases duran más tiempo, pero las horas de trabajo dedicadas son menores, debido a un mayor número de asignaturas (5) que compaginar con el trabajo en comparación al segundo cuatrimestre (2).

7. ENTORNO SOCIOECONÓMICO

En este apartado se hace una estimación de los recursos necesarios para llevar a cabo el proyecto y su coste. Para esta estimación se toman en consideración costes de personal (recursos humanos) y costes materiales (principalmente hardware y software).

7.1. Costes de personal

Para llevar a cabo la estimación de costes de personal, es necesario realizar una estimación de horas. Para ello, se parte de la planificación del proyecto del capítulo anterior.

El trabajo empieza el día 25 de septiembre con la primera reunión con el tutor y se distribuye en los dos cuatrimestres que dura el curso 2017/2018. Realizando un desglose de horas por etapas del proyecto se pueden establecer costes más precisos.

Se definen 4 roles diferentes para llevar a cabo el proyecto. En este trabajo solo ha existido la participación del alumno, por lo tanto, será el alumno la única persona que desempeñe los roles que se definen.

- Jefe de Proyecto / Diseñador: Se encarga de las labores de dirección, además, se encarga de la investigación y definición de la arquitectura de la aplicación.
- Analista: Su labor es especificar requisitos a partir de las necesidades del cliente y definir la estructura básica del sistema.
- Programador: Se encarga de la codificación y pruebas de la aplicación.
- Documentador: Su trabajo consiste en generar el informe del trabajo.

A continuación, se realiza el desglose de horas del proyecto. En las siguientes tablas se pueden ver las fases del proyecto, el rol desempeñado en cada fase, el número de días que dura la fase y el número de horas trabajadas.

7.1.1. Planteamiento

La fase de planteamiento coincide con el primer cuatrimestre del curso, por lo que solo se dedica una hora al día de trabajo.

Tabla 32: Horas de trabajo. Planteamiento

Planteamiento			
Fase	Rol	Días	Horas
Introducción	Jefe de Proyecto	26	26
Estado del Arte	Jefe de Proyecto	54	54
Análisis	Analista	48	48
TOTAL		128	128

7.1.2. Desarrollo

Con el inicio de la fase de desarrollo empieza el segundo cuatrimestre del curso, por lo tanto, se dispone de más tiempo para trabajar, en este punto, se dedican dos horas al día de trabajo.

Tabla 33: Horas de trabajo. Desarrollo

Desarrollo			
Fase	Rol	Días	horas
Prototipo 1	Programador	27	54
Análisis	Analista	7	14
Prototipo 2	Programador	41	82
TOTAL		75	150

7.1.3. Documentación

La fase de documentación se desarrolla a lo largo de todo el proyecto, en esta fase se dedica una hora al día de trabajo a la elaboración de la memoria.

Tabla 34: Horas de trabajo. Documentación

Documentación			
Fase	Rol	Días	Horas
Redacción	Documentador	149	149
Revisión	Jefe de Proyecto	49	49
TOTAL		198	198

7.1.4. Costes de personal totales

Por último, el cómputo de horas totales y los costes de cada rol.

Tabla 35: Horas de trabajo. Totales

Totales (Horas)	
Fase	Horas
Planteamiento	128
Desarrollo	150
Documentación	198
TOTAL	476

Tabla 36: Costes de personal. Totales

Totales (Costes)			
Rol	Horas	Sueldo (€/h)	Coste (€)
Jefe de Proyecto/Diseñador	129	50	6450
Analista	62	35	2170
Programador	136	30	4080
Documentador	149	30	4470
Total	476		17170

7.2. Costes materiales

La parte restante de los costes se reduce a costes hardware, costes software y material fungible. Las siguientes tablas detallan los diferentes costes materiales.

7.2.1. Costes Hardware

Entre los costes hardware se encuentra un ordenador portátil dónde se ha llevado a cabo el desarrollo la aplicación y la elaboración de la memoria. Por otra parte, para probar la aplicación en diferentes plataformas y formatos se ha necesitado de dos tablets con distintos sistemas operativos, una tablet Android y una tablet iOS. Además, para probar la responsividad de la aplicación, se ha requerido también de un smartphone. Los productos concretos y su precio se detallan en la siguiente tabla.

Tabla 37: Costes hardware

Costes Hardware	
Elemento	Precio sin IVA (€)
Portátil Sony VAIO	489.85
Tablet Samsung Galaxy Tab A 10.1	155.37
Apple iPad	288.43
Smartphone OnePlus 3	289.26
Total	1222.91

7.2.2. Costes Software

Los costes software se componen de todos aquellos recursos software que han sido necesarios para el desarrollo del trabajo. En la siguiente tabla se detallan los recursos.

Tabla 38: Costes software

Costes Software	
Elemento	Precio sin IVA (€)
Windows 10	119.83
MS Office 365	57.02
Atom Editor	Gratis
Cuenta Github (5 meses)	35.00
Total	211.85

7.3. Resumen de costes y totales

Todas las cifras manejadas hasta el momento son precios antes de impuestos. A los costes de personal es necesario aplicar costes de IRPF y Seguridad Social (24% sobre el total). A los gastos materiales es necesario aplicarles el 21% del Impuesto sobre Valor Añadido (IVA).

En la siguiente tabla aparecen reflejados todos los costes, con sus respectivos impuestos y el coste total del proyecto.

Tabla 39: Costes totales

Resumen de costes			
Elemento	Costes Sin Impuestos (€)	Impuestos (€)	Coste (€)
Costes de personal	17170.00	4120.80	21290.80
Costes Materiales (Hardware y Software)	1434.76	301.30	1736.06
Costes Hardware	1222.91	256.81	1479.72
Costes Software	211.85	44.49	256.34
Total			23026.86

En resumen, el coste total del proyecto es de **23026.86 Euros**.

8. CONCLUSIONES

En este apartado se vuelcan las conclusiones abstraídas sobre los resultados de este TFG. Primero se realizarán observaciones respecto a los objetivos planteados y a los resultados obtenidos, después, se realizará una discusión sobre las expectativas y el trabajo futuro.

Al comenzar este trabajo se plantea el problema que tienen los educadores en la elaboración de anticipadores con pictogramas y en la importancia que tienen estos anticipadores en el día a día de los niños. Como solución, se desarrolla una aplicación que debe sustituir a las herramientas actuales que los educadores usan. Esta aplicación, además, debe de ser fácil de usar. Para ello, se hace uso de la metodología de diseño centrado en el usuario y se preparan reuniones con el usuario final en las que el usuario final prueba la aplicación y puede proponer cambios y alternativas.

Los resultados de las reuniones son positivos, a través de la reacción del usuario final y los comentarios que este hace, se puede decir que la aplicación cumple con los objetivos planteados en términos de facilidad de uso y el propósito de ser una herramienta específicamente preparada para la realización de planificadores con pictogramas.

La aplicación ha podido ser probada por un educador, pero, para poder obtener información más fiable y de carácter más general, sería necesario contar con más probadores de la aplicación y estudiar los resultados durante un mayor periodo de tiempo. Esto no ha podido ser posible debido a las limitaciones de la envergadura TFG.

De cara al futuro, el objetivo es continuar con el desarrollo de la aplicación en un ámbito más cercano a la investigación, por medio de un Trabajo de Fin de Máster. La idea es llevar la aplicación a un ámbito más colaborativo entre profesores y padres y estudiar los resultados en la vida de los niños, padres y educadores.

Para concluir, el desarrollo de esta aplicación y del trabajo en general debe entenderse como un compromiso hacia la sociedad. En este trabajo son más importantes los efectos de la tecnología en la vida de los educadores, padres y niños que la propia tecnología en sí. Es la utilización de la ingeniería informática como medio para resolver un problema presente en la sociedad actual.

9. RESUMEN EN INGLÉS. ABSTRACT.

9.1. Introduction

Children with Autism Spectrum Disorders (ASD) often find difficulties with routine changes and transitioning between activities. This generates a strong dependency on their caregivers (mainly educators and parents).

There are visual supports like activity schedulers that help to decrease these dependencies on other people. The main issue regarding this kind of support it is that, it requires a long time for educators to elaborate them.

In the case of ASD classrooms (classrooms specially prepared for kids with ASD), activity schedulers are a useful resource to define the daily classroom activities that compose a student's routine.

Educators make use of these resources to increase student's understanding about past, present and future tasks. This way, they enhance a more independent behavior amongst their students. However, there are some exceptions with non-routine activities like school trips. Non-routine activities are problematic and generate atypical behaviors amongst children with ASD, researchers point out behaviors such as hitting, biting, crying and screaming (Banda and Grimmet, 2008).

To reduce the impact of these non-routine activities on kid's behavior and keep their stress levels to minimum, educators develop another kind of visual schedulers, they call those "anticipators" or "anticipations". Anticipators are visual schedulers with pictograms and text, these schedulers include information about how, when, and with whom a non-routine activity will be developed. With these anticipators, educators get to reduce kid's atypical behaviors when they face activities that are out of their daily routine. Educators show the anticipators to the kids at least two weeks before the day of the activity so they can prepare for it. Alongside the anticipator itself, some educators prepare a secondary product that includes the most representative pictograms in a little accordion of pictures, this way an educator can guide his students at every moment when the day of the activity comes.

The main issue with these anticipators is that, educators need to spend a lot of time on them since they don't have a specific creation tool for them to use. To create the anticipator, educators use Microsoft Word. Later this document, there is an analysis of the process of creating a scheduler using this program. On the other hand, to develop the secondary support product, the picture accordions, they must use paper and scissors and

a lot of patience and time. Furthermore, parents may not be able to develop their own anticipators for their kids because of the technological barrier and a lack of experience in the creation of these.

Now that we have seen some of the problems educators have to deal with and the context of these problems, we can explain the main objectives of this work.

The main objective of this work is to develop an easy-to-use application that educators and parents can use to create visual schedulers. The application also aims to be used as a replacement for the picture accordions mentioned earlier. Finally, the application would allow the users to export and share their own creations.

Creating an easy-to-use application implies that educators will invest less time in the creation of the anticipators, and, since the anticipators can be shared, it can promote a more collaborative environment between parents and educators. With a tablet or smartphone application, it is possible to break the barriers that parents may find in the creation of this content. Parents will be able to use the application without a deep technical or technological knowledge. This way they will be able to help their kids with the content created in the app.

9.2. State of the Art

The purpose of this section is to give a general vision about all the work that has been done until now through the existent literature on this field. The objective of this section is to identify and synthesize the main current ideas on the field and set a starting point for this work.

First, it is important to define what exactly are the Autism Spectrum Disorders. It is a range of various neurobiological disorders which main symptoms are “persistent deficiencies in communication and social interaction as well as repetitive and restrictive patterns of behavior, interests or activities” (American Psychiatric Association, 2013). This disorder affects to around three hundred thousand people, from which around fifty thousand are kids in school age (Europapress, 2016).

Communication skills are vital to be able to involve in society and develop the ability to interact with other people. From these problems to communicate arises the need for the development of support products and augmentative and alternative communication systems.

Support products and Augmentative and Alternative Communication Systems.

Support products are defined as “any product especially made to prevent, compensate, control, mitigate or neutralize deficiencies, activity limitations or participation restrictions” (UNE-EN ISO 9999, 2011). Among the most used support products for children with cognitive disabilities or ASD there are devices such as communication boards, electronic communicators and software for computers or mobile devices.

On the other hand, Augmentative and Alternative Communication Systems are communication alternatives to spoken language that aim to improve or compensate for the deficiencies suffered by people with disabilities. In these kind of communication systems, symbol systems are used with support products (ARASAAC, 2018).

A pictographic system is a good example of alternative and augmentative communication system. Pictographic systems are flexible and can be adapted to a wide range of people with various levels of cognitive disability.

Visual Schedulers

A visual scheduler is a pictographic system composed by a series of ordered pictograms. These pictograms can be presented in various ways, for example, in strips of pictograms of variable length or individually one by one. In addition, the pictograms may be accompanied with text.

As we can read in Banda and Grimmer (2008), the use of this kind of visual supports is a promising educational strategy. There are some principles by which visual schedulers work. These principles also treated in the mentioned article.

- People with ASD prefer visual stimuli over the rest since it is their main source of information.
- People with ASD tend to prefer the use of objects over receiving orders from a person.

There are articles such as the one written by Takanori Koyama & Hui-Ting Wang (2011) in which an analysis of visual planners is made. Visual schedulers achieve to promote a more independent and self-determined behavior. In addition to this, there are other benefits that can be observed with the use of visual schedulers:

- Visual schedulers can adapt to different types of people with different cognitive disabilities, they are not exclusive to people with ASD.

- Visual schedulers can be used both at school and at home.
- They are applicable to different areas and topics such as games, school activities, home chores, etc.

In both mentioned articles it is said that, visual schedulers achieve the purpose of reducing the dependency of people with ASD on other people like their educators, parents or caregivers.

Technological point of view Presupuesto





From a technological perspective, this kind of visual support can be integrated in mobile devices such as smartphones and tablets. There is another article written by Davies, Stock and Wehmeyer (2002) in which they analyze the benefits of the use of electronic devices with people with cognitive disabilities. It increases the independency and self-determination and reduces the need of external cues to accomplish tasks on people with cognitive disabilities.

The mentioned article was written in 2002, in 2018 we can find a lot of autism-related applications. Next, an analysis of the most recommended and used mobile applications can be found. The objective of this analysis is to extract both positive and negative features that are present on the analyzed apps.

The analyzed applications were: Araword, PictogramAgenda, CPA and LetMeTalk. All of these are popular applications with ratings above 4 stars on the Google PlayStore.

The following table summarizes the main characteristics found in the applications under analysis.

Tabla 40: Comparative between apps

				
Works in smartphones/tablets	Yes	Yes	Yes	Yes
Has a tutorial	No	No	Yes	No
Uses ARASAAC pictograms	Yes	Yes	Yes	Yes
Allows to use personal images	Yes	Yes	Yes	Yes
Creations can be shared	Yes	Yes	No	No
Easy-to-use design	Yes	No	No	Yes
Has a pictogram search-bar	No	No	Yes	Yes
Has a secuencia mode	No	Yes	Yes	Yes
Allows to select key pictograms	No	No	No	No
Uses a color communication system	Yes	No	Yes	Yes

This summary table is useful to draw different conclusions, such as, for example, the importance of the common features in all the analyzed apps.

The application should work both in tablets and smartphones.

The application should also make use of the ARASAAC pictogram collection. Another important feature is to allow users to use their own personal images as pictograms.

From this table, it is striking that not all applications allow the user to share their creations in a standard format such as pdf or jpeg.

Another important point is that, the design of all four applications has a lot of room to improve, in some apps it is obvious that the needs of the end user were not considered.

The pictogram finder should be an essential tool in this kind of applications where many images are handled.

Lastly, the possibility of adding significance to the visual schedulers with colors is a characteristic that should be considered.

Implementation alternatives

Nowadays there are multiple ways to build a mobile application, so it is important to carry out a study of the existing alternatives. The technology used to develop the final product is less important than the utility of the product itself, but it is important to know the pros and cons of the existing alternatives.

Native Code

One way to develop the application would be through native code for a given platform. For Android, you could choose to program in Java or Kotlin and for iOS to program in Objective-C or Swift.

The main advantage would be the performance, the main problem is that to make an application that works on both platforms, the application must be programmed twice.

Hybrid Application (Cordova, Ionic)

The next alternative would be to develop a hybrid application. Hybrid apps are classic web applications composed of HTML, CSS and Javascript that are contained in a WebView.

A WebView works on both iOS and Android. The application would be the same as a native application but cordova runs a small server on which the web application runs.

The advantages of using a framework such as, for example, ionic, is that there are many already created components that can be used to save time. On the other hand, the developed application would work the same on both Android and iOS.

The main disadvantage is that the performance is lower than the native code.

Compiled to native code App (React native)

Finally, there is a third alternative that is to develop with a framework like React Native, which allows the development of cross-platform apps with javascript. The difference with this and a hybrid app is that the code is compiled into native code.

It is not 100% cross-platform, there are components that only work for a certain platform, and, in a certain way, you must program the application more than once. As a downside, it is a relatively new framework and still has some aspects to be improved.

9.3. System Analysis

The objective of this sections is to carry out a system analysis taking as a starting point the state of the art already analyzed. In addition, given that this work intends to be based on the User-Centered Design (UCD) methodology, it is essential to identify the end users to know what their capabilities and needs are. The objective is to develop an application to solve specific needs and have a high usability for the end users.

Users Identification

The main groups of users to which the application is directed are the following.

- Educators in ASD classrooms.
- Parents of children with cognitive impairments.
- People with cognitive deficits capable of learning how to make their own visual schedulers with pictograms.

The main group of users to which the application is directed are the educators in ASD classrooms since they need to provide their students daily with these visual supports. Making these visual schedulers is part of their routine and, above all, the daily routine of their students. Educators are forced to invest a lot of time in the development of these and, as we have seen, there is not an existing application that can be used to prepare these anticipators, so they must use general purpose tools.

On the other hand, the application is aimed at parents, parents who do not have the technical capabilities to develop these visual schedulers on their own. The currently available applications are not easy to use, and they do not offer enough help to carry out a visual scheduler that can be interpreted satisfactorily by a child with ASD.

In addition, this application would be directed to any person with ASD with the ability to learn to develop their own visual schedulers with pictograms, this case is mentioned in the article by Takanori Koyama & Hui-Ting Wang (2011).

The main need for parents and people with ASD would be to make an application accessible and easy to use.

Finally, we must not forget that the objective of this project is to improve the quality of life of children with ASD in special education classes. This is intended to be achieved through the visual schedulers created by their caregivers, but they are also final users of

the application, since they will be those who make use of the content that will be created with this application.

From the state of the art and the problems already defined, one can begin to define the needs and difficulties that educators must face on their work.

Among all the problems that educators have, they emphasize onto being forced to use general purpose tools, which entails a loss of time and a bad experience due to the impediments of these tools.

System definition

The system to develop is an application that will work on mobile devices, specifically smartphones and tablets.

Now it is necessary to define the most basic and essential features of the application.

- Allow the creation of visual planners with pictograms.
- Make use of the ARASAAC pictogram repository.
- Allow the addition of own images and photos as pictograms.
- Include a pictogram finder.
- Include in the application a way of visualizing the created schedulers, where the user can see the whole sequence of pictograms of the scheduler.
- Allow the users to share their own creations in pdf format.

The next step in this work is to define software requirements for subsequent development of the application. All the software requirements, functional and non-functional can be found in chapter 3.5 of this document.

Since the application is developed following the UCD methodology, there will be meetings with the end user. This means that the software requirements will change with the feedback received from the end user after testing the prototypes.

9.4. System Implementation

For the implementation of the application a prototyping process has been followed. Starting with a non-functional prototype, followed by two prototypes in ionic. Upon completion of both prototypes, the application has been presented to the end user.

The results of the meetings were both positive and both resulted in good evaluations by the educator. In addition, from these meetings the educator was able to propose new ideas and some changes.

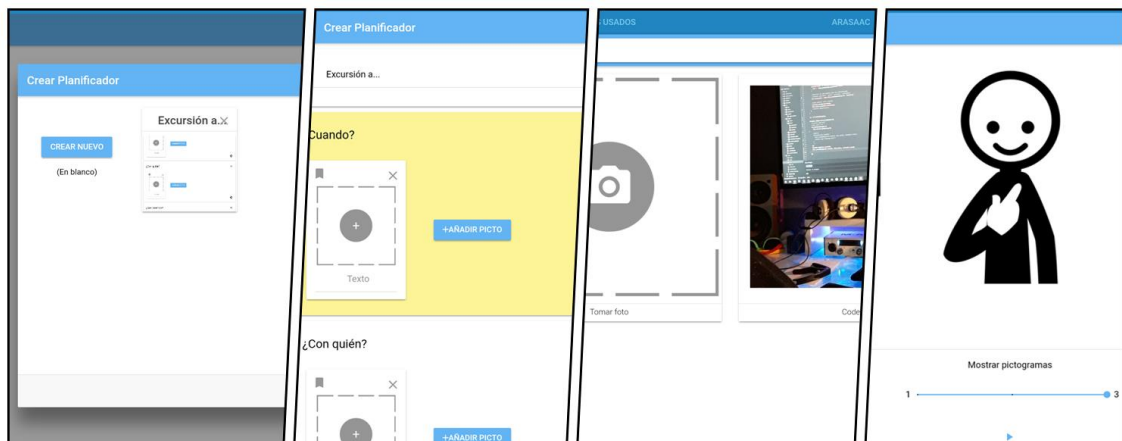


Ilustración 38: Ionic final prototype screenshots

In section 4 of this document you can read a detailed explanation of all the screens that make up the application and the proposed changes between the iterations of the prototypes.

Budget and resources

The resources necessary for the development of this project are divided into hardware resources and software resources. An estimation can be made based on the project planification, the cost of the defined roles and the estimation of hours of work needed.

Tabla 41: Project budget

Project budget			
Item	Tax free price (€)	Taxes (€)	Price (€)
Staff costs	17170.00	4120.80	21290.80
Material Costs (Hardware & Software)	1434.76	301.30	1736.06
Hardware	1222.91	256.81	1479.72
Software	211.85	44.49	256.34
Total			23026.86

A more detailer view of the costs can be seen in section 7 of this document.

9.5. Conclusions

In this section the abstracted conclusions on the results of this work are included. First, there will be some observations regarding the objectives and the results obtained, then a discussion will be held on expectations and future work.

Objectives

At the beginning of this work, the problems that educators had to face in the elaboration of anticipators with pictograms and the importance of these anticipators in children's day-to-day was presented. As a solution, an application has been developed, and its purpose is to replace the current tools that educators use nowadays. This application also had to be easy to use. To achieve this, the user-centered design methodology has been used. Meetings were held with the end user and the results were positive.

In general terms and taking into account the duration of the project, it could be said that the proposed objectives and the expected results were met.

Future work

The application was tested by an educator, but to obtain more reliable and more general information regarding the application, it would be necessary to have more testers of the application and study the results for a longer period of time. This could not be possible due to the limitations of time and resources of this work.

Looking to the future, the goal is to continue with the development of the application, this time from a research perspective, through a master's thesis. The idea is to take the application to a more collaborative environment between teachers and parents and study the results in the lives of children, parents and educators.

To conclude, the development of this application and work in general should be understood as a commitment towards society. In this work, the effects of technology in the lives of educators, parents and children is more important than the technology itself.

10. BIBLIOGRAFÍA

- Banda, D. R., & Grimmer, E. (2008). Enhancing social and transition behaviors of persons with autism through activity schedules: A review. *Education and Training in Developmental Disabilities*, 324-333.
- Koyama, T., & Wang, H. T. (2011). Use of activity schedule to promote independent performance of individuals with autism and other intellectual disabilities: A review. *Research in developmental disabilities*, 32(6), 2235-2242.
- Davies, C. (2008). Using visual schedules: A guide for parents.
- American Psychiatric Association. (2013). *Diagnostic and statistical manual of mental disorders (DSM-5®)*. American Psychiatric Pub.
- Europapress.es (4 abril, 2016) Alrededor de 50.000 niños en edad escolar tienen autismo en España. Recuperado de <http://www.europapress.es/epsocial/infancia/noticia-alrededor-50000-ninos-edad-escolar-tienen-autismo-espana-20120402170938.html>.
- UNE-EN ISO 9999:2007 (septiembre, 2007). Productos de apoyo para personas con discapacidad., Clasificación y terminología.
- Davies, D. K., Stock, S. E., & Wehmeyer, M. L. (2002). Enhancing independent task performance for individuals with mental retardation through use of a handheld self-directed visual and audio prompting system. *Education and Training in Mental Retardation and Developmental Disabilities*, 209-218.
- Google (30 mayo, 2018) Material Design Guidelines. Recuperado de <https://material.io/design/>
- Snow, K. (2009). People first language. *Disability is Natural*, <http://www.disabilityisnatural.com/peoplefirstlanguage.htm> (accessed June 4, 2007).
- Abras, C., Maloney-Krichmar, D., & Preece, J. (2004). User-centered design. *Bainbridge, W. Encyclopedia of Human-Computer Interaction. Thousand Oaks: Sage Publications*, 37(4), 445-456.

11. GLOSARIO

ARASAAC: Portal Aragonés de Comunicación Aumentativa y Alternativa

CAA: Comunicación Aumentativa y Alternativa

LOPD: Ley Orgánica de Protección de Datos de Carácter Personal

SAAC: Sistemas Aumentativos y Alternativos de Comunicación

TEA: Trastornos del Espectro Autista

MVC: Modelo Vista Controlador

TFG: Trabajo de Fin de Grado

UCD: User-Centered Design

PFL: People-First Language

PECS: Sistema de Comunicación por el Intercambio de Imágenes

ANEXO I. Consentimiento Informado para la entrevista.

Consentimiento Informado para Participantes de Investigación

El propósito de esta ficha de consentimiento es proveer a los participantes en esta investigación con una clara explicación de la naturaleza de esta, así como de su rol en ella como participantes.

La presente investigación es conducida por **Diego Alejandro González Ordóñez**, de la **Universidad Carlos III de Madrid**. La meta de este estudio es obtener información respecto a los problemas y las necesidades del usuario final de la aplicación en desarrollo para el Trabajo de Fin de Grado: Desarrollo de un Planificador Visual Interactivo Para Niños con Discapacidad Cognitiva en Clase de Educación Especial.

Se le pedirá responder preguntas en una entrevista. Esto tomará aproximadamente 30 minutos de su tiempo. Lo que conversemos durante esta sesión se grabará, de modo que el investigador pueda transcribir después las ideas que usted haya expresado.

La información que se recoja será confidencial y no se usará para ningún otro propósito fuera de los de esta investigación.

Si tiene alguna duda sobre este proyecto, puede hacer preguntas en cualquier momento durante su participación en él. Igualmente, puede retirarse del proyecto en cualquier momento sin que eso lo perjudique en ninguna forma. Si alguna de las preguntas durante la entrevista le parece incómoda, tiene usted el derecho de hacérselo saber al investigador o de no responderlas.

Desde ya se le agradece su participación.

Acepto participar voluntariamente en esta investigación, conducida por Diego Alejandro González Ordóñez. He sido informado de que la meta de este estudio es obtener información respecto a los problemas y las necesidades del usuario final de la aplicación en desarrollo para el Trabajo de Fin de Grado: Desarrollo de un Planificador Visual Interactivo Para Niños con Discapacidad Cognitiva en Clase de Educación Especial.

Me han indicado también que tendré que responder cuestionarios y preguntas en una entrevista, lo cual tomará aproximadamente 30 minutos.

Reconozco que la información que yo provea en el curso de esta investigación es estrictamente confidencial y no será usada para ningún otro propósito fuera de los de este estudio sin mi consentimiento.

He sido informado de que puedo hacer preguntas sobre el proyecto en cualquier momento y que puedo retirarme del mismo cuando así lo decida, sin que esto acarree perjuicio alguno para mi persona. De tener preguntas sobre mi participación en este estudio, puedo contactar a **Diego Alejandro González Ordóñez** al teléfono **697321172**.

Entiendo que una copia de esta ficha de consentimiento me será entregada, y que puedo pedir información sobre los resultados de este estudio cuando éste haya concluido. Para esto, puedo contactar a **Diego Alejandro González Ordóñez** al teléfono anteriormente mencionado.

ALBERTO HERNÁNDEZ JIMÉNEZ



1-3-2018

Nombre del Participante

Firma del Participante

Fecha